

Řízení robota Lego NXT pomocí headsetu EPOC

Lego NXT Robot Control using EPOC Headset

Zadání bakalářské práce

Student:

Jan Beseda

Studijní program:

B2647 Informační a komunikační technologie

Studijní obor:

2612R025 Informatika a výpočetní technika

Téma:

Řízení robota Lego NXT pomocí headsetu EPOC
Lego NXT Robot Control using EPOC Headset

Jazyk vypracování:

čeština

Zásady pro vypracování:

Cílem práce je realizace řízení robota Lego Mindstorms NXT pomocí headsetu EPOC. Emotiv EPOC je neuroheadset sloužící jako rozhraní pro interakci člověka s počítačem. Pro toto zařízení byl na katedře informatiky vyvinut SDK, který bude použit k realizaci řízení robota.

1. Seznamte se s headsetem EPOC.
2. Seznamte se robotem Lego Mindstorms NXT.
3. Seznamte se s SDK vytvořeným na katedře informatiky.
4. Naimplementujte software pro řízení NXT pomocí EPOC.
5. Navrhněte experimenty, realizujte a vyhodnoťte je.

Seznam doporučené odborné literatury:

- [1] Vourvopoulos A., Liarokapis F. Brain-controlled NXT Robot: Tele-operating a robot through brain electrical activity <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5962098>
- [2] <http://mindstorms.lego.com>
- [3] <http://www.emotiv.com>

Dále podle pokynů vedoucího bakalářské práce.

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **RNDr. Eliška Ochodková, Ph.D.**

Datum zadání: 01.09.2014

Datum odevzdání: 28.04.2016



doc. Dr. Ing. Eduard Sojka
vedoucí katedry



prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě dne 27. dubna 2016



.....

Rád bych poděkoval těm, kteří, ať už nepřímo svými publikacemi či svými konzultacemi, mi byli nápomocni. Zvlášť děkuji paní RNDr. Elišce Ochodkové, Ph.D., která mi velmi pomohla svými radami a vedla mě tím správným směrem k úspěšnému dokončení této bakalářské práce. Dále pak panu Ing. Petru Gajdošovi, Ph.D. za konzultace k Emotiv Headsetu a jeho SDK a Vysoké škole báňské – Technické univerzitě Ostrava, která mi vypůjčila potřebné zařízení k realizaci tohoto projektu.

Abstrakt

Cílem této práce je realizace řízení robota Lego Mindstorms EV3 pomocí Emotiv headsetu EPOC. Lego Mindstorms EV3 je řídicí jednotka, která ovládá jednotlivé senzory a motory připojené k ní. Emotiv neuroheadset EPOC slouží jako rozhraní pro interakci člověka s počítačem. Byly vytvořeny 2 aplikace. Jedna, MindBot RobotPart pro obsluhu robota a jeho ovládání, která běží na řídicí jednotce EV3, a další MindBot Emotiv2Robot, která slouží jako grafické prostředí a slouží jako konektor mezi headsetem EPOC a EV3 aplikací. Následně byly provedeny experimenty, které testovaly jednotlivé aspekty ovládání robota pomocí headsetu.

Klíčová slova: headset, robot, Emotiv, Lego, Mindstorms, EPOC, EV3, EEG, rozhraní pro interakci člověka s počítačem, C++, Qt, Java, VŠB

Abstract

Aim of this Paper is realisation of controlling the robot Lego Mindstorms EV3 by Emotiv headset EPOC. Lego Mindstorms EV3 is control unit which controls the connected particular sensors and motors. Emotiv neuroheadset EPOC serves as brain-computer interface. Two applications has been implemented. One MindBot_RobotPart for taking care of control of the robot which runs on EV3 control unit and second MindBot Emotiv2Robot which serves as GUI and connector between EPOC headset and EV3 application. At the end, experiments has been conducted which tested particular aspects of controlling the robot by the headset.

Keywords: headset, robot, Emotiv, Lego, Mindstorms, EPOC, EV3, EEG, Brain-computer interface, C++, Qt, Java, VŠB

Seznam použitých zkratk a symbolů

SDK	– Software development kit
GUI	– Graphical user interface
VŠB	– Vysoká škola báňská
EEG	– Electroencephalography
USB	– Universal Serial Bus
CD	– Compact disc
CMS	– Common Mode Sense
DRL	– Driven Right Leg
ADC	– Analog-to-digital converter
Hz	– Hertz
LSB	– Least significant bit
AC	– Alternating current
API	– Application programming interface
SD	– Secure Digital
CRATEL	– Center for Research in Arts, Technology, Education, and Learning
EMG	– Elektromyografie
HMI	– Human-machine interface
B	– byte
ANSI	– American National Standards Institute
ot/min	– Otáček za minutu
bit/s	– Bit za sekundu
Ncm	– Newtoncentimetr
IDE	– Integrated Development Environment
GPS	– Global Positioning System
MOC	– Meta-Object Compiler
ARM	– Advanced RISC Machine

Obsah

1	Úvod	5
2	Podobné technologie a jejich využití	6
2.1	Brain-controlled NXT Robot	6
2.2	Mind-controlled R/C Car	6
2.3	EPOC-alypse Mind Controlled Car	7
2.4	Steering a Tractor by means of an EMG-Based Human-Machine Interface .	7
3	EEG	8
3.1	Mozkové vlny	9
3.2	Brodmannovy oblasti	10
4	Použité zařízení	12
4.1	Headset EPOC	12
4.2	Lego Mindstorms	15
5	Architektura robota	18
5.1	Návrh	18
5.2	Realizace	18
6	Aplikace	19
6.1	Analýza	19
6.2	Návrh	19
6.3	Implementace	20
6.4	Postup používání	23
7	Experimenty	26
7.1	Různé jazyky	26
7.2	Vtělená kognice	26
7.3	Odezva mezi aplikacemi	26
7.4	Experiment přesnosti a odezvy při ovládání robota headsetem	27
8	Závěr	29
9	Reference	30
	Přílohy	32
A	Obrázky	33
B	Tabulky	42
C	Výpisy kódu	45

Seznam tabulek

1	Typy mozkových vln a jejich charakteristika. [19]	9
2	Tabulka zobrazující přehled nezbytného hardwaru, softwaru a znalostí . .	24
3	Tabulka specifikací headsetu. [5]	44

Seznam obrázků

1	Hlavní laloky a brázdy mozkové kůry.[3]	9
2	Mezinárodní 10 - 20 systém rozmístění elektrod EEG headsetu, kde je označeno 14 kanálů snímající aktivitu mozku: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8 a AF4. [4]	10
3	Předozadní zobrazení mozkové kůry s vyznačenými Broadmannovými oblastmi [21].	11
4	Komponenty přístroje Emotiv EPOC: 1. neuroheadset, 2. USB přijímač, 3. instalační CD, 4. hydratační obal s 16 elektrodami, 5. solný roztok a 6. USB kabel. [5]	13
5	Ukázka diagramu aktivit pro používání API, ke komunikaci mezi EmoEngine a vlastní implementací. [8]	14
6	Konfigurace NXJ pluginu řídící jednotky Lego Mindstorms EV3 v Eclipse IDE.	17
7	Abstraktní schéma možného propojení signals a slots pro vytvoření událostmi řízené architektury [28].	21
8	Správné nasazení headsetu [7].	24
9	Úvodní obrazovka aplikace MindBot Emotiv2Robot.	25
10	Cheprer, model robota, z pohledu ze shora.	34
11	Cheprer, model robota, z pohledu z pravého boku	35
12	Cheprer, model robota, z pohledu zespodu. Na konstrukci jsou zeleně vyznačeny EV3 Large Servo Motor a modře EV3 Medium Servo Motor. . . .	35
13	Aplikace Emotiv Control Panel - Cognitiv Suite s připojeným headsetem a bez nasazení.	36
14	Diagram případu užití zobrazuje případ, kdy aktér, uživatel headsetu, trénuje jednotlivé kognitivní akce.	37
15	Digram případu užití ukazuje aktéra, uživatele headsetu, jak ovládá robota za pomoci aplikace Emotiv2Robot a aplikace RobotPart.	38
16	Třídní diagram aplikace MindBot Emotiv2Robot.	39
17	Třídní diagram aplikace MindBot RobotPart.	40
18	Aplikace Emotiv TestBench zobrazuje správný či špatný kontakt elektrod s hlavou uživatele a EEG aktivitu [7].	41
19	Tabulka zobrazující výsledky experimentu.	43

Seznam výpisů zdrojového kódu

1	Deklarování a ovládání motorů robota	16
2	Deklarace slotu a jeho napojení na implementovaný signál ve třídě QTimer.	21
3	Implementace funkce zatočení ve třídě Executor.	22
4	Implementace hlavní metody třídy MainController. Je volána každých 100ms.	45

1 Úvod

Ovládat robota pomocí myšlenek? V dnešní době se nejedná o fikci, ale o realitu společně s dalšími technologiemi, jako je například ovládání kybernetického obleku pomocí bioelektrických signálů vyslaných mozkem do svalů, které se vyskytují na kůži. Pomocí senzorů uvnitř nasazeného obleku dokáže oblek vypomáhat lidem, kteří mají problémy s pohybem, nebo dokonce nemohou chodit. Díky zážitku z pohybu si mozek zapamatuje tyto signály a dokáže pomocí nich překonat dočasné ochrnutí. Ovládání robota myšlenkami je problematika řešená v rámci této práce, přesněji řečeno přenášení mozkových signálů získaných z headsetu a posílání do ovládací jednotky robota.

Téma interakce člověka s přístrojem se stává čím dál více rozšířené. Technologie jako virtuální realita a EEG headsety jsou příkladem tohoto narůstajícího trendu. Jediným limitem v dalším vývoji nových technologií je naše představivost a co dokážeme vytvořit a propojit. Proto jsem si zvolil toto téma řízení robota pomocí EEG headsetu, abych se zařadil mezi skupinu lidí, zabývajících se touto problematikou a abych nabyté zkušenosti a znalosti mohl v budoucnu využít k vytvoření něčeho nového, a posunout tak tuto stále dosud neprobádanou problematiku o krok dál.

Cílem této práce bylo prozkoumat možnosti propojení robota s člověkem a následně implementovat program, který dokáže ovládat řídicí jednotku robota Lego Mindstorms NXT pomocí EEG headsetu EPOC od firmy Emotiv. V průběhu vypracovávání této práce přišla na trh nová verze řídicí jednotky Lego Mindstorms EV3. Je výkonnější a otevírá nové možnosti ovládání a komunikace, které byly využity v implementaci aplikace této práce. Proto došlo ke změně řídicí jednotky z NXT na EV3. Experimenty, které budou testovat koncentraci člověka a přesnost ovládání, jsou nedílnou součástí projektu vedoucí k zjištění, zda by aplikace mohla sloužit jako nástroj pro zvýšení koncentrace, trénování představivosti nebo být součástí léčebné terapie.

V kapitole Podobné technologie a jejich využití 2 jsou uvedeny podobné projekty, které byly použity jako inspirace k vypracování tohoto projektu. V následující kapitole EEG 3 jsem se zabýval studiem EEG, mozkových vln a Broadmannových oblastí. Dále 4 jsou popsána zařízení, které jsem použil v tomto projektu, EPOC Headset a Lego Mindstorms EV3. Původně zadané zařízení Lego Mindstorms NXT je taky uvedeno, společně s důvodem pro rozhodnutí změny zařízení. V kapitole 5 je popsán design robota a jeho možnosti pohybu. V kapitole Aplikace 6 jsou popsány 2 aplikace, od analýzy až po implementaci a návod k použití. MindBot Emotiv2Robot se spouští na PC a komunikuje s EPOC headsetem. Umožňuje trénování kognitivních akcí, použitých k pohybu robota, a přeposílá zpracovaná data po síti, které přijímá druhá aplikace MindBot RobotPart. MindBot RobotPart běží na EV3, zpracovává přijatá data a vysílá příkazy jednotlivým motorům, které umožňují pohyb robota. Pro zjištění úspěšnosti této práce byly provedeny experimenty 7. Zaměřují se na přesnost a odezvu při ovládání robota pomocí EEG headsetu.

2 Podobné technologie a jejich využití

EEG Headset EPOC od firmy Emotiv, který byl uveden na trh v roce 2009, nebyl prvním headsetem svého druhu. Měl své předchůdce a má přibývajících následovníky. Jeden z prvních EEG headsetů je MindSet od firmy NeuroSky, vydán v roce 2007, který má pouze jednu elektrodu umístěnou v oblasti čelního laloku. MindSet dokáže snímat 2 psychické stavy a mrknutí očí. Začátkem roku 2014 Emotiv uvedlo novou verzi headsetu Insight, která je zjednodušenou verzí headsetu EPOC, která se skládá z 5 elektrod a slouží k prozkoumání každodenního života a pozorování mozkové aktivity při sportu.

V posledních letech vzniklo několik podobných projektů, které využívají Emotiv EPOC headset pro ovládání pohybu robota 2.1, RC auta 2.2 2.3, a dokonce i traktoru 2.4. Díky těmto projektům se dostáváme z předchozí fikce našich představ do reality, nebo se jí přibližujeme. Zároveň mi pomohly při vymýšlení implementace pro tento projekt a jakou variací úkonů robota ovládat.

2.1 Brain-controlled NXT Robot

Brain-controlled NXT Robot je projekt vytvořen dvojicí studentů z Coventry University. Tento projekt umožňoval ovládat robota NXT pomocí EEG headsetu NeuroSky a Emotiv EPOC. Na řídicí jednotce robota Lego Mindstorms NXT běžel Java Virtual Machine, LeJOS. Tudíž program, který obsluhoval přijímání signálů ze sítě Bluetooth, mohl být napsaný v programovacím jazyku Java, v Eclipse IDE. Při použití headsetu NeuroSky se robot pohyboval na bílém navigačním plánu a jeho funkce byly jet dopředu a zastavit. Pro příkaz jet dopředu bylo třeba zvýšit úroveň pozornosti uživatele, a když pozornost klesla pod danou úroveň, robot zastavil. Design robota obsahoval světelný senzor NXT, a když robot na navigačním plánu najel na okrajovou černou hranici, světelný senzor tuto změnu zaznamenal a program byl ukončen. Na druhou stranu při použití headsetu EPOC, který dokáže snímat také kognitivní akce, se robot pohyboval 4 směry: dopředu, dozadu, doleva a doprava. Pro natrénování těchto kognitivních akcí pro daného uživatele byla použita aplikace, dodávaná v rámci Emotiv Development Kit, Emotiv Control Panel. Na počítači běžel program, který snímal stisknutí jednotlivých kláves a podle nich posílal robotovi příkazy k pohybu. Pro namapování stisknutí kláves a kognitivních akcí zaznamenaných headsetem byla použita aplikace Emotiv EmoKey [27].

2.2 Mind-controlled R/C Car

Projekt Mind-controlled R/C Car byl proveden v Center for Research in Arts, Technology, Education, and Learning (CRATEL) na Wichita State University. Zpočátku se skupina, podílející se na tomto projektu, pokusila využít NeuroSky headsetu, ale zjistila, že technologie a data přijímané headsetem nebyly dostačující pro tento projekt. Zjistili, že NeuroSky headset mohl číst pouze mozkové vlny z jedné strany mozku. Z tohoto důvodu bylo velmi obtížné získat použitelné a přesné údaje. Poté, co zjistili tento fakt, rozhodli se použít Emotiv EPOC headset z důvodu pokročilejšího vybavení a přesnějšího čtení. Pro fungování tohoto prototypu bylo potřeba zkušeného uživatele, který dokáže

udržet koncentraci jedné kongnitivní akce nad úrovní 0,599, na škále od 0 do 1. Jakmile úroveň koncentrace byla vyšší než 0.599, signál byl zpracován a poslán na dálkové ovládání. Směr, ve kterém se vůz pohyboval, byl určen polohou hlavy uživatele. Například, aby auto jelo dopředu, uživatel musel naklonit hlavu vzhůru. Když si uživatel přál zastavit auto, snížil úroveň koncentrace a zaklonil hlavu do středové polohy [26].

2.3 EPOC-alypse Mind Controlled Car

EPOC-alypse Mind Controlled projekt byl vytvořen 4člennou skupinou studentů na University of Central Florida. Používali Emotiv EPOC headset pro ovládání RC auta. Komunikace probíhala přes mini-ITX základní desku a vlastně sestavenou Arduino desku, která byla vložena do karosérie RC auta. Auto se dokázalo pohybovat 4 směry, přičemž pro pohyb auta dopředu, dozadu a zastavení byly použity kongnitivní akce push, pull a neutral. Pro ovládání směru byla použita mimika. Pro zatočení doleva bylo použito mrknutí levým okem a pro zatočení doprava mrknutí pravým okem. Při vybírání sady akcí, které bude uživatel headsetu trénovat pro kongnitivní akce, využili TestBench, aplikaci od firmy Emotiv, pro zobrazení snímaných EEG dat z jednotlivých elektrod [17].

2.4 Steering a Tractor by means of an EMG-Based Human-Machine Interface

Elektromyografie(EMG) je human-machine interface(HMI), které slouží jako komunikační kanál mezi člověkem a strojem, který funguje na získávání a zpracovávání signálů EMG. Tento projekt zkoumá využití HMI založené na EMG signálech pro řízení zemědělských traktorů. Pro tento účel byl použit EPOC headset. EPOC, pomocí 14 senzorů, měří a zpracovává EMG a elektroencefalografické(EEG) signály z pokožky hlavy řidiče. Byla použita detekce čtyř natrénovaných svalových událostí z pokožky hlavy řidiče. Kombinace pohledu očí doprava a otevření čelisti, pohled očí doprava a zavření čelisti, pohled očí doleva a otevření čelisti a pohled očí doleva a zavření čelisti. Řízení pomocí EMG bylo srovnáváno s manuálním řízením a s autonomní navigací pomocí GPS. Výsledky při testování tohoto řešení ukázaly, že je možné řídit traktor pomocí HMI, založeného na EMG signálech, s téměř stejnou přesností jako při ručním řízení [25].

3 EEG

Brain-computer interface je přímé komunikační spojení mezi mozkiem a externím elektronickým zařízením. BCI většinou slouží k asistenci při kognitivních nebo smyslově motorických úkonů. Od roku 2007 se začala rozšiřovat BCI zařízení dostupné na trhu. Tato zařízení snímají elektrickou aktivitu mozku a pomocí svých knihoven dokážou přeložit nasnímané nezpracované EEG data do jednotlivých kognitivních, pocitových a expresivních akcí. Jedním z prvních úspěšných zařízení byl headset od firmy NeuroSky, který má jeden senzor na čelním laloku. Další senzor headsetu Emotiv EPOC se skládá ze 7 párů senzorů a 2 referencí. Firma Emotiv vydala v roce 2015 nový headset Insight, který má už jen 5 senzorů, ale jeho účel je pro snímání EEG signálů při sportovních aktivitách, oproti headsetu EPOC, který je zaměřen více na výzkum a umožňuje tím lepší snímání kognitivních myšlenek u člověka.

Elektroencefalografie (EEG) je diagnostická metoda používána k záznamu elektrické aktivity mozku. EEG se řadí mezi neinvazivní metody, na rozdíl od invazivní metody elektrokortikografie (ECoG), která snímá signály přímo z mozkové kůry. Používá se při neurochirurgických zákrocích kvůli přesnosti. U EEG dochází k oslabení signálu při průchodu přes lebku, řádově v mikrovoltech [18]. EEG měří kolísání napětí, vznikající z toků iontového proudu z mozku. Tyto toky iontového proudu jsou udržovány neurony a získávají svůj náboj z membránových transportních proteinů, které fungují jako jednoduchá čerpadla, přenášející ionty přes cytoplazmatické membrány. Headset zachytává tyto hodnoty, když jeden neuron uvolní velké množství iontů. Ionty můžou tláčit na jiné neurony, ty zase na další neurony a tento proces se opakuje. Tomuhle jevu říkáme objemová komunikace, a když tato vlna dosáhne elektrody na EEG detektoru, vyvine sílu na kov uvnitř každé elektrody. Tento rozdíl v tlaku a uvolnění na kov mezi dvěma elektrodami se zaznamenává jako EEG. [17]

Emotiv EPOC EEG headset má 14 senzorů, které jsou dotekové přijímače na základě solného roztoku. Každý senzor je ideálně namapovaný a výhodně umístěný ve 14 různých oblastech a lalocích mozku pro optimální měření. Všechny těchto 14 laloků a oblastí je rozděleno do specifických regionálních oblastí. Každá z nich má různé funkční aspekty.

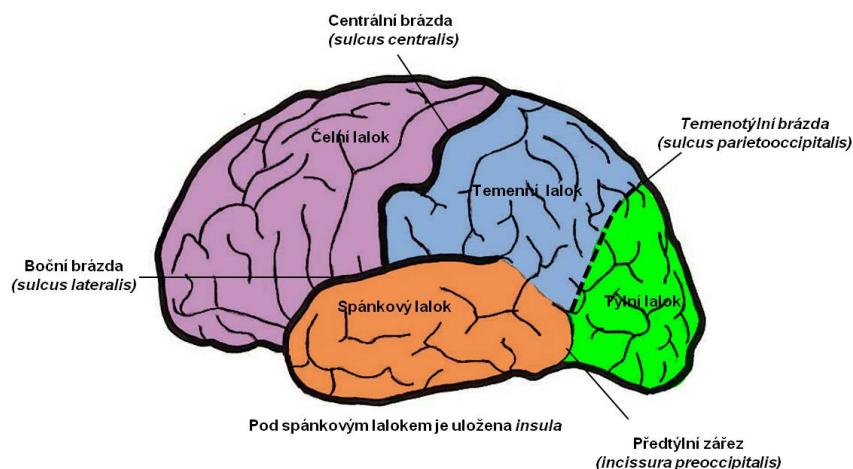
Oblasti mozkové kůry:

Čelní(frontální) lalok

- Řízení motorických funkcí, a to od jejich plánování až po vlastní provedení
- Řečová funkce
- Krátkodobá paměť

Temenní(parietální) lalok

- Vnímání tělesných vjemů - chuť, dotyk, teplo, chlad, bolest a polohocit



Obrázek 1: Hlavní laloky a brázdy mozkové kůry.[3]

Temporální lalok

- Sluch a paměťová schopnost

Okcipitální(týlní) lalok

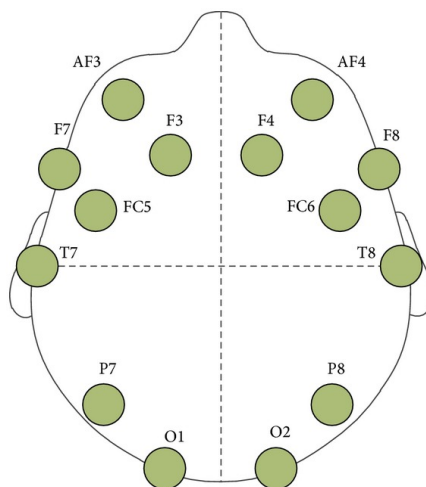
- Vnímání a zpracování zrakových podnětů

3.1 Mozkové vlny

Jsou to neuronové oscilace lidského mozku. 14 kanálů znázorněných na obrázku2, snímá aktuální vlnovou aktivitu mozku, ty pak dále můžeme rozdělit do hlavních frekvenčních pásem: Alfa, Beta, Gamma, Delta a Theta. V tabulce 1 jsou uvedena relevantní frekvenční pásma, jejich frekvence, oblast převahy a stav mysli, při které se projevují.

Typ rytmu	Rozsah frekvence (Hz)	Oblast převahy nebo maxima	Podmínky přítomnosti
Alfa	8-12	okcipitální a parietální	bdění, relaxace, zavřené oči
Beta	18-30	precentrální a frontální	bdění, motorický klid
Gamma	30-50	precentrální frontální	bdění
Delta	0,5-4	variabilní	ospalost, bdění
Theta	5-7	frontální a temporální	bdění, vzrušení nebo stres

Tabulka 1: Typy mozkových vln a jejich charakteristika. [19]



Obrázek 2: Mezinárodní 10 - 20 systém rozmístění elektrod EEG headsetu, kde je označeno 14 kanálů snímající aktivitu mozku: AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8 a AF4. [4]

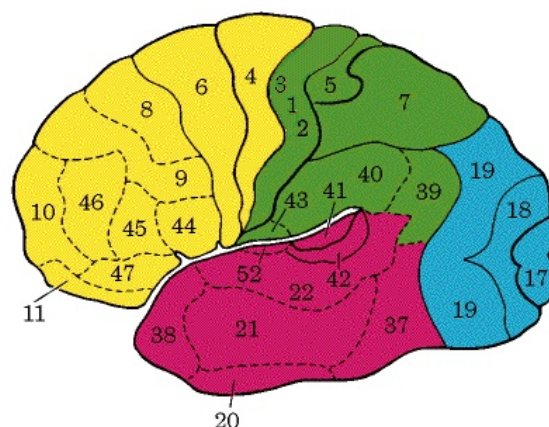
Zmiňované typy mozkových vln jsou užitečné pro trénování a nalezení správných myšlenek, na které se zaměřit. Avšak hodnoty jednotlivých typů mozkových vln nejsou použity pro čtení EEG a ovládání robota. Hodnoty, které zaznamenává EEG, jsou snímány na oblastech párů senzorů, které jsou rozmístěny po lebce. Tyto oblasti jsou nazývány Brodmannovy oblasti.

3.2 Brodmannovy oblasti

Rozděluje mozkovou kůru podle typů buněk a jejich struktury do tzv. Brodmannovy mapy. Ta obsahuje 11 krajín a 52 okrsků. Rozdíly ve stavbě kůry souvisí s jejím funkčním zapojením. Rozlišujeme primární korové oblasti (projekční) a sekundární korové oblasti. Na obrázku 3 je schéma, na kterém můžeme vidět jednotlivé rozdělení oblasti a kde se nacházejí v mozku. [20]

Headset EPOC má rozložení svých elektrod nejvíce nad 7 okrsky Broadmannovy mapy. Níže máme seznam jednotlivých okrsků a jejich označení dle Broadmannovy mapy v závorce za názvem.

- Dorzolaterální prefrontální asociační kortex(9) a Přední prefrontální asociační kortex(10) má na starost planování komplexního kognitivního chování, projev osobnosti, rozhodování, úsudek, předvídání a sociální chování.
- Čelní okohybné pole(8) zajišťuje koordinaci očních pohybů. Spravuje souhyby očí, hlava a krk ve směru viděného předmětu.
- Chuťová oblast(43)



Obrázek 3: Předozadní zobrazení mozkové kůry s vyznačenými Broadmannovými oblastmi [21].

- Střední spánkový gyrus(21), společně s okrsky 20, 22, 37 a 38, zajišťuje rozeznávání obličejů a předmětů.
- Primární motorická oblast(4) zodpovídá za ovládání distálních svalů končetin.
- Asociační somatosenzitivní oblast(7) vytváří prostorové mapy okolí pro orientaci v prostoru a motorické plánování pohybu.

4 Použité zařízení

4.1 Headset EPOC

Neuroheadset Emotiv EPOC byl vytvořen australskou společností Emotiv Systems, založenou roku 2003, která se zabývá vývojem rozhraní, propojujícího mozek s počítačem pomocí EEG technologie. Původní zakladatelé jsou neurovědec prof. Allan Snyder, čipový designer Neil Weste a podnikatelé v technologiích Tan Le a Nam Do [1].

4.1.1 Specifikace

EPOC se skládá s 14 elektroencefalogramových senzorů a 2 referencí. Byl vyvinut převážně pro zábavu, průzkum trhu, snímání EEG aktivity a neuroterapii. Pomocí API a detekčních knihoven dodávaných s headsetem je možné lehce snímat a zobrazit frekvenční pásma EEG signálu společně s pocity, jako jsou vzrušení, dlouhodobé nebo okamžité, frustrace, meditace, zábava a nuda. Kromě citových vjemů, které jsou snímány pomocí headsetu, je možné monitorovat výrazy obličeje, jako jsou pohled vlevo, pohled vpravo, zamrknání, mrknutí pravým či levým okem, zvednutí obočí, zatnutí zubů a úsměv. Pomocí gyroskopického senzoru v neuroheadsetu je možné snímat pohyby hlavy. Podrobnější specifikace headsetu je uvedena v tabulce 3.

4.1.2 Příslušenství

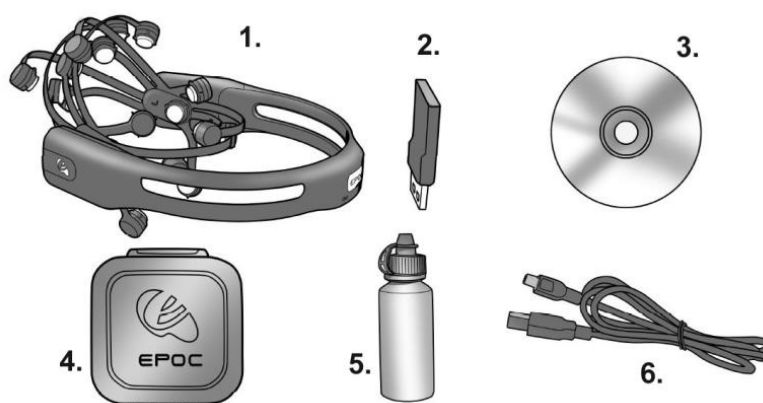
Pro správné použití neuroheadsetu je třeba správně navlhčit jednotlivé elektrody pomocí solného roztoku. Je možné použít čistou vodu, ale protože jednotlivé elektrody jsou metalické, dochází ke korozi a vytvoření zeleného povlaku na jednotlivých elektrodách a snižuje se tak životnost elektrod. Pro dosažení co nejlepšího snímání a prodloužení životnosti elektrod je zapotřebí využít solný roztok, případně vodivý gel.

Čtená data jsou přenášena z headsetu do počítače přes USB přijímače, pomocí Bluetooth. Jakmile je navázáno spojení, headset vyšle 32B zprávu, která je šifrovaná pro jazyk Emokey, ty jde pak dešifrovat pomocí AES. Emokit oddělí gyroskopická data a aktuální data ze senzorů a odešle je.

Pro nabití EPOC headsetu prosím připojit headset do USB portu počítače pomocí kabelu dodávaného s headsetem EPOC. Při nabíjení není možné používat headset pro snímání dat. Headset je nabitý, jakmile přestane svítit dioda červeně, ta se rozsvítí po zapojení headsetu, přes USB kabel, do počítače. Vyobrazení příslušenství můžete vidět na obrázku 4.

4.1.3 Emotiv Nástroje

Emotiv dodává v rámci programu Emotiv Control Panel, který můžeme vidět na obrázku 13, 3 nástroje: Expressiv Suite, Affectiv Suite a Cognitiv Suite. Každý z nich používá jiný druh interpretace signálů ze senzorů pro zobrazení různého druhu zaznamenaných dat.



Obrázek 4: Komponenty přístroje Emotiv EPOC: 1. neuroheadset, 2. USB přijímač, 3. instalační CD, 4. hydratační obal s 16 elektrodami, 5. solný roztok a 6. USB kabel. [5]

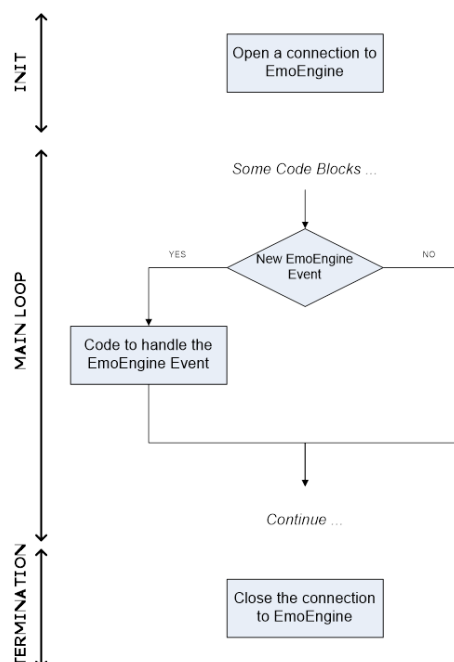
Kromě Emotiv Control Panel aplikace Emotiv dodává s headsetem program EmoComposer, který slouží k simulování jednotlivých signálů, které dokáže vyslat headset. Simulace se dají vyvolat pomocí grafického rozhraní, nebo je načíst ze souboru, který byl vygenerován aplikací Test Bench, která slouží pro vyobrazení a zaznamenání EEG signálů z jednotlivých senzorů. Poslední aplikací, kterou Emotiv dodává v Developers edition, je EmoKey, který dokáže přiřadit natrénované akce k stisknutí tlačítka klávesnice či myši.

V první verzi projektu byl použit Emotiv Control Panel pro natrénování jednotlivých kognitivních akcí a pro načtení profilu. Později však správa profilu a trénování akcí bylo implementováno v rámci aplikace MindBot Emotiv2Robot, která komunikuje s headsetem. Pro rozhodnutí, jakou skupinu vybraných kognitivních akcí použít pro ovládání robota, pomohla aplikace Test Bench, která zobrazovala přímý přenos signálů testovaných akcí, a potom už stačilo porovnávat, který druh akcí byl nejvíce rozdílný pro zlepšení přesnosti ovládání a rozpoznání natrénovaných kognitivních akcí.

4.1.4 Emotiv SDK

Součástí Emotiv SDK je Emotiv API, které je známé jako ANSI C rozhraní, které je deklarované třemi hlavičkovými soubory, obsahujícími rozhraní. `Edk.h`, který slouží ke komunikaci mezi externími aplikacemi a EmoEngine. `EmoStateDLL.h` obsahuje konstanty a rozhraní pro zachytávání jednotlivých akcí a stavů. `EdkErrorCode.h` definující návratové hodnoty EDK API. Implementace zmíněných hlavičkových souborů je obsažena ve 2 Windows DLL souborech (`edk.dll` a `edk_utils.dll`).

4.1.4.1 EmoEngine Nedílnou součástí SDK je zmiňovaný Emotiv EmoEngine, který referuje logickou abstrakci funkcí v `edk.dll`. EmoEngine komunikuje s Emotiv headsetem skrze USB přijímač a zpracovává EEG signály ze 14 kanálů, které jsme si popsali v předchozí sekci. Jejich rozložení a názvy můžeme vidět na obrázku 2. EmoEngine do-



Obrázek 5: Ukázka diagramu aktivit pro používání API, ke komunikaci mezi EmoEngine a vlastní implementací. [8]

stává gyroskopická data ze senzoru polohy umístěném v headsetu. EmoEngine dokáže spravovat profily a uživatele, tedy připojené headsety. Zpracovaná data se dají získat pomocí třídy EmoState. Funkce, které dokáží upravit, případně získat EmoEngine nastavení, jsou s prefixem EE_. EmoState je třída, obsahující funkce pro zjištění současného stavu, po provedené poslední detekci. Jsou to expresivní, pocitové a kognitivní stavy. Data jsou získávána funkcemi s prefixem ES_. Ukázku použití API vidíme na obrázku 5.

4.1.5 Rozšířené SDK

Dekódování EEG signálu není jednoduchou záležitostí. Signál je velmi slabý a může být rušen podněty, jako je mrknutí oka nebo nepatrný pohyb prstu. Neexistuje zatím žádný univerzální vzor EEG signálů, který by dokázal přesně určit, že tato nervová aktivita je příkaz k zatnutí svalů na rameni a upažení pravé paže. Řešení je namapovat několik náhodně výběrových souborů dat a provést například Deanův a Dixonův test. Další postupem je využít algoritmy, jako jsou Neural Networks nebo Support Vector Machines.

Za tímto účelem je vyvíjeno rozšířené SDK na Vysoké škole báňské Technické univerzitě Ostrava pod vedením Ing. Petrem Gajdošem, Ph.D. SDK, které v této fázi již umožňuje lepší využití EEG EPOC headsetu, společně s Qt aplikací, která zahrnuje základní funkce EDK. Další součástí je základní koncept GPU knihovny pro manipulaci s daty [9].

4.2 Lego Mindstorms

Prvním modelem je programovatelná kostka RCX. Obsahuje 8bitový H8/300 mikrokontroler jako vnitřní procesor, který má pouze 32K RAM pro uložení firmware a programů od uživatele. Nahrání napsaného programu z počítače na RAM je prováděno přes infračervené rozhraní. Původním programovacím jazykem byl jazyk Brick Logo. Měl první vizuální programovací prostředí LEGOsheets, vytvořené na University of Colorado v roce 1994. Od té doby Lego, společně s MIT, ušlo velký kus cesty a vytvořilo další úspěšné modely, jako jsou Lego Mindstorms NXT a Lego Mindstorms EV3.

4.2.1 Lego Mindstorms EV3

Je nejnovější řídicí jednotkou řady Lego Mindstorms EV3. Jeho úspěšný předchůdce Lego Mindstorms NXT měl být původně použit. V průběhu zpracovávání projektu vyšel model EV3 a v následující sekci je popsáno, proč se nakonec použil tento model proti původně zadanému NXT.

4.2.2 Srovnání a specifikace

Důvodem pro použití řídicí jednotky Lego Mindstorms EV3, byl lepší maximální výkon a přenosová rychlost. Tímto se eliminovaly potenciální limity výkonu použité technologie. Zároveň, pro většinu předchozích prací, byla použita řídicí jednotka NXT, čímž tomuhle projektu přibyl další rozdíl v implementaci a možnost experimentování s novější technologií.

Pro srovnání, model EV3 má USB host port, do kterého můžeme připojit WiFi USB adaptér. V tomto projektu byl použit WiFi USB adaptér Edimax N150 s přenosovou rychlostí až 150 Mbit/s. Bylo vytvořeno připojení pomocí WiFi, namísto spojení přes Bluetooth, s maximální přenosovou rychlostí 720 kbit/s, jako je tomu u modelu NXT. EV3 má 4 porty pro ovládání motoru oproti modelu NXT, který má 3. Micro SD slot s maximální kapacitou karty 32GB, který umožňuje snadné nahrání nového firmware. NXT běží na proprietárním operačním systému, zatímco EV3 na Linuxu. Procesor ARM9 je šestkrát výkonnější u EV3(300 MHz), než má NXT(48 MHz) procesor Atmel 32-Bit ARM7. Velikost RAM EV3 je 64 MB a NXT 64 KB [10].

4.2.3 Příslušenství

Byla použita sada Education EV3 Core Set, která obsahuje: EV3 programovatelnou řídicí jednotku(1x), EV3 Large Servo Motor(2x), EV3 Medium Servo Motor(1x), dotykový senzor(2x), senzor pro detekci barvy(1x), gyroskopický senzor(1x), ultrazvukový senzor(1x), propojovací kabely, USB kabel pro spojení řídicí jednotky s počítačem, dobíjecí baterie a jednotlivé kousky stavebnice, pro vytvoření konstrukce robota. Z této sady byl vytvořen model robota, který byl pojmenován Cheprer. Kromě stavebnice sady EV3 byly použity 2 velké motory, 1 střední motor a ultrazvukový senzor.

4.2.3.1 EV3 Large Servo Motor Výkonný motor s 160-170 ot/min a točivým momentem 20 Ncm při pohybu a 40 Ncm ze stání. Jeho rozměry jsou 112x56x40mm [15].

4.2.3.2 EV3 Medium Servo Motor Motor používaný k otáčení lehčích konstrukcí, s točivým momentem 8 Ncm při pohybu a 12 Ncm ze stání. EV3 Medium Servo Motor je rychlejší než EV3 Large Servo Motor, až 250 ot/min. Je velice skladný s rozměry 24x32x72mm a proto v některých případech a požadavcích na design, využitelnější než velký EV3 servo motor [15].

4.2.3.3 EV3 Ultrasonic Sensor Ultrazvukový senzor generuje zvukové vlny a čte jejich ozvěny pro zjištění a změření vzdálenosti od objektů. Může taky vyslat jednoduché zvukové vlny a fungovat jako sonar nebo odposlouchávat a čekat na zvukovou vlnu, která spustí program. Měřená vzdálenost může být od 1 cm do 250 cm, s přesností na 1 cm [15].

4.2.4 Software

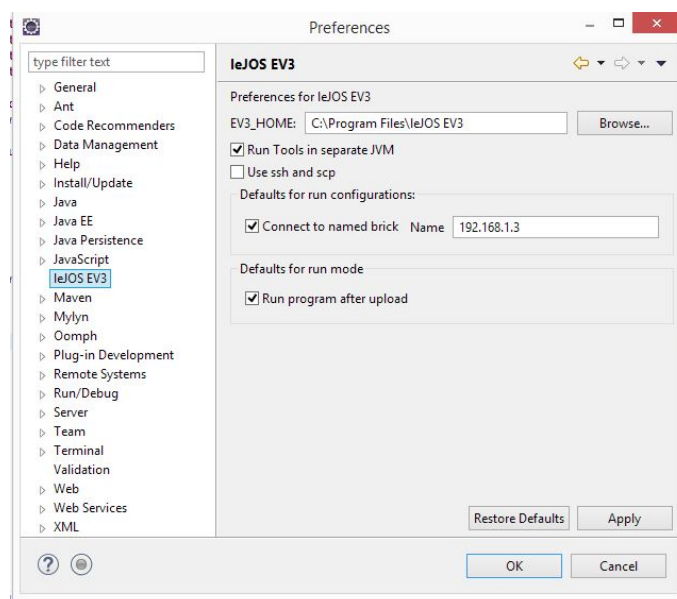
Od roku 1994 a prvního programovacího jazyka Brick Logo byla vytvořena spousta dalších prostředí, SDK a firmware. V podstatě v dnešní době lze naprogramovat NXT nebo EV3 pomocí většiny známých jazyků, jako jsou: C, C++, Python, Perl, Java, Objective-C, Ruby, MATLAB a Ada. Všechny mají své výhody a zaměření, ke kterému účelu jaký software zvolit. Například pokud plánujete pracovat s větším objemem dat a provádět nad nimi matematické operace, je dobré využít RWTH – Mindstorms NXT Toolbox, který využívá MATLAB. Pro programování v MAC OS je možné použít LegoNXTRemote, který využívá Objective-C. Pro tento projekt byl zvolen firmware LeJOS, nahrazující původní firmware EV3 programovatelné kostky.

4.2.5 LeJOS

LeJOS obsahuje Java Virtual Machine, který umožňuje řídicímu modulu spouštět programy v programovacím jazyku Java. LeJOS je převážně používán pro výukové účely na školách, ale jeho knihovny podporují širokou škálu funkcí, které dokáží pracovat s většínou Lego Mindstorms senzorů. Pomocí leJOS knihoven lze pracovat s GPS, rozpoznáním hlasu a mapovacích technologií. Může být propojen s Eclipse IDE nebo se spouštět přímo z příkazové řádky.

LeJOS je ukázkový příklad využití objektově orientovaného jazyka, skrývající implementaci senzorů a motorů za několik rozhraní, a tím zajistit dokonalé zapouzdření. Díky tomuto ukázkovému zapouzdření dokáže vývojář pracovat na vysoké úrovni abstrakce a nemusí se strachovat o hexadecimální adresy jednotlivých komponentů.

```
EV3LargeRegulatedMotor motorLeft = new EV3LargeRegulatedMotor(MotorPort.B);
EV3LargeRegulatedMotor motorRight = new EV3LargeRegulatedMotor(MotorPort.C);
motorLeft.rotate(angleL);
motorRight.rotate(angleR);
```



Obrázek 6: Konfigurace NXJ pluginu řídicí jednotky Lego Mindstorms EV3 v Eclipse IDE.

Výpis 1: Deklarování a ovládání motorů robota

LeJOS plugin znázorněný na obrázku 6 dostupný pro Eclipse IDE a NetBeans, který podporuje práci s API, zvanou NXJ. Tímto je umožněno jednoduché kompilování, spouštění a nahrávání programu přímo do robota z vývojového prostředí. [11]

5 Architektura robota

Bylo třeba se zamyslet, co od robota očekávat a jaké jsou naše schopnosti se soustředit na danou kognitivní nebo expresivní akci, jak získané data z headsetu, přeložené EmoEngine, dokážeme zpracovat a pomocí řídicí jednotky EV3 vyjádřit.

5.1 Návrh

Základní vzhled robota byl velice jednoduchý a vycházel z architektury robota Lego® Mindstorms® Education EV3 Core set, který je poháněn dvěma velkými motory EV3. Splnil však účel a dokázalo se na něm provést několik experimentů a pohyby robota dopředu, dozadu, doprava a doleva byly díky tomuto modelu implementovány. Po hlubším zamyšlení a poznání Emotiv SDK byla provedena kompletní rekonstrukce kostry robota a rozšířena o motor EV3 střední velikosti, který dokáže provádět přesnější pohyby s menší reakční dobou. Motor slouží k otáčení ultrazvukového senzoru, který má tvar očí a je použit jako hlava robota.

5.2 Realizace

Robot byl složen na základě kostry robota Spik3r, navrženého Lego® Mindstorms®.[14] K dispozici nebyly všechny díly, které model Spik3r využívá, a proto bylo třeba improvizovat. Díky prvotnímu plánu ovládat robota pomocí řídicí jednotky NXT, byly k dispozici stavební díly i této sady, které jsou se sadou EV3 kompatibilní. Motory a senzory sady řídicí jednotky NXT dokážou komunikovat s EV3 řídicí jednotkou, ale nebyly potřeba, protože základní Education EV3 Core sada obsahuje všechny požadované motory a senzory pro sestavení a funkčnost modelu. Výsledný model robota byl pojmenovaný Cheprer, který můžeme vidět na obrázcích 10 a 11.

5.2.1 Použité součástky a jejich propojení

Byly použity 3 servo motory a umístěny mezi základy kostry modelu Cheprer, jak můžeme vidět na obrázku 12. Otáčení hlavy robota bylo docíleno za pomoci 2 ozubených kol, které mají vzájemnou polohu os 90° . Pro zpomalení pohybu a zvýšení přesnosti je průměr hnacího kola menší než kola hnaného. Nohy byly vytvořeny pouze jako pohyblivé části, které neslouží k pohybu robota, ale spíše pro vytvoření co nejpřesnější autentičnosti s broukem vrubounem, který byl použit jako vzor pro vzhled robota. Byly připojeny na hřídel jednotlivých kol, levého a pravého, a při jejich otáčení se jednotlivé nohy rozpohybují.

6 Aplikace

6.1 Analýza

Aplikace by měla sloužit k propojení řídicí jednotky robota EV3 a headsetu EPOC. Přijímat signály zaznamenané headsetem, rozpoznat je a poslat na EV3 řídicí jednotku, na které bude běžet program, který bude mít připravené funkce pro rozpoznání kódu s instrukcemi k provedení akce. Knihovny pro komunikaci s headsetem jsou napsány v C a C++, proto bude použito aplikace, která bude založena na jazyku C. Aplikace se nebude muset řešit "wrapperem" pro jiný programovací jazyk. Vybrané knihovny a firmware pro robota jsou založeny na jazyku Java. Bude tedy nutné zprostředkovat komunikaci mezi 2 rozdílnými programovacími jazyky.

Vzniknou dva programy. Jeden, který bude běžet na počítači a bude napsán v programovacím jazyce C a C++. Bude mít funkce, jako je trénování kognitivních akcí pro pohyb robota a posílání zakódované zprávy po síti přes WiFi druhému programu, který poběží na řídicí jednotce EV3. Program, který poběží na řídicí jednotce EV3, bude přijímat zakódovanou zprávu ze sítě, přeloží ji a pošle signál motorům, aby vykonaly akci z rozkódované zprávy. Program, který bude běžet na počítači, se bude jmenovat MindBot Emotiv2Robot a program, který poběží na robotovi, se bude jmenovat MindBot RobotPart.

6.2 Návrh

6.2.1 Technická specifikace

Podkladem pro technologická rozhodnutí byly podobné předchozí projekty, uvedené v sekci 2. Dále publikace uvedené v referencích. Pro implementaci obou aplikací byly použity níže uvedené technologie.

- C, C++ programovací jazyky použité pro aplikaci MindBot Emotiv2Robot.
- Qt knihovna, pomocí které bylo vytvořeno grafické rozhraní aplikace MindBot Emotiv2Robot.
- Emotiv Development Kit API použita pro komunikaci s EmoEngine v rámci aplikace MindBot Emotiv2Robot.
- JavaSE-1.7(jre1.8.0_25) programovací jazyk, ve kterém byla napsána aplikace MindBot RobotPart.
- LeJOS v0.9.1 firmware, který byl nahrán na řídicí jednotku EV3, který umožňuje běh Java Virtual Machine na EV3. Bylo tedy možné použít programovací jazyk Java pro aplikaci MindBot RobotPart.

6.2.2 Funkční specifikace

Před implementací aplikací bylo třeba definovat jednotlivé funkce aplikací a vztahy mezi nimi. Za tímto účelem byly vytvořeny diagramy případu užití.

Diagram případu užití na obrázku 14 zobrazuje případ, kdy aktér, uživatel headsetu, trénuje jednotlivé kognitivní akce. Aplikace Emotiv2Robot pak odpovídá na jednotlivé akce uživatele. Nejprve je třeba navázat spojení s headsetem a vybrat nebo vytvořit si profil, který bude sloužit jako model natrénovaných kognitivních akcí. Poté bude mít uživatel možnost si vybrat, jakou kognitivní akci chce trénovat a Emotiv2Robot ho bude informovat o stavu tréninku, jestli byl úspěšný nebo ne.

Další diagram případu užití 15 ukazuje aktéra, uživatele headsetu, jak ovládá robota za pomoci aplikace Emotiv2Robot a aplikace RobotPart. Propojení z headsetem je prvním krokem. Aby uživatel mohl začít odesílat kódovanou zprávu, která obsahuje příkaz k pohybu, je zapotřebí mít natrénované všechny kognitivní akce, proto při stisknutí tlačítka pro ovládání robota je provedena kontrola natrénovaných akcí. Aplikace RobotPart po spuštění čeká na příchozí zprávy z aplikace Emotiv2Robot na otevřeném kanálu, takže není třeba provádět žádnou další interakci s aplikací RobotPart. RobotPart po přijetí zprávy zprávu dekóduje a přeložený příkaz k pohybu přiřadí jednotlivým motorům na základě vnitřní logiky aplikace RobotPart. Po přiřazení úkonů jednotlivým motorům aplikace RobotPart dá příkaz k jejich vykonání.

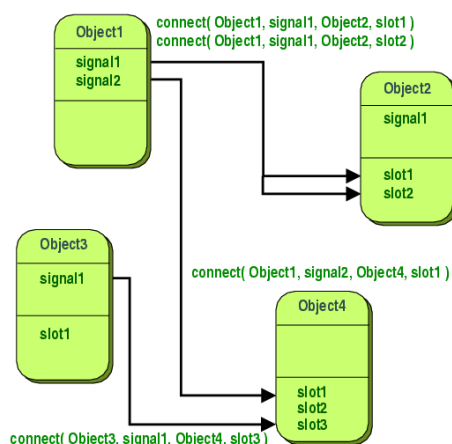
6.3 Implementace

6.3.1 MindBot Emotiv2Robot

Aplikace MindBot Emotiv2Robot je naprogramována v jazyce C a C++ s použitím frameworku Qt, napsaného v programovacím jazyce C++. Qt umožňuje psát aplikace v událostmi řízené architektuře, za pomoci signals a slots. Signals jsou funkce, které jsou spojeny s událostí, či změnou stavu objektu, která je propojena s funkcí deklarovanou jako slot. Schéma, jak může vypadat takové spojení, můžeme vidět na obrázku 7.

Podobný druh komunikace je možný skrze callback. Callback je ukazatel na funkci. Takže pokud chcete zpracovávající funkci, aby vás upozornila na nějakou událost, je jí třeba předat ukazatel na jinou funkci, tedy callback. Funkce, která zpracovává nějakou událost, potom volá callback v případě potřeby. Callback má však 2 zásadní nedostatky. První, callback není typově bezpečný. Nikdy si nemůžeme být jisti, že funkce, která zpracovává událost, zavolá callback se správnými argumenty. Druhý, callback je pevně spojený s funkcí, která zpracovává událost, protože tato funkce musí vědět, který callback volat. Tyto 2 nedostatky jsou v Qt vyřešeny právě díky signals a slots. Podpis funkce definované jako signal se musí shodovat s podpisem funkce deklarované jako slot, tím je zajištěna typová bezpečnost. Signals a slots jsou volně vázané. Třída, která volá signal neví, které funkce definované jako slot přijímají tento signal. Tím je vyřešen druhý problém v případě použití callback.

Qt obsahuje preprocesor MOC (Meta-Object Compiler), který umožňuje označení funkcí jako signals a slots. Před kompilací MOC analyzuje zdrojové soubory napsané v Qt rozšířeném C++ a vygeneruje C++ zdrojové soubory, které odpovídají standardu.



Obrázek 7: Abstraktní schéma možného propojení signals a slots pro vytvoření událostmi řízené architektury [28].

Emotiv2Robot je implementován několika třídami. Tyto třídy se dají rozdělit do 3 skupin. První skupina tříd obsahuje kontrolery a jednoho manažera. Každý kontroler je zaměřený svými funkcemi na získávání určitého typu dat z EPOC headsetu. *ProfileController* je třída, která obsahuje funkce pro práci s profilem. Dokáže profil vytvořit, či vybrat s existujících a přiřadit jej připojenému headsetu. Také ukládá profil s natrénovanými kognitivními akcemi do externího souboru *profile.bin*. Samotné ukládání a načítání natrénovaného profilu je implementováno ve třídě *ExternalProfileManager*. Třída *CognitiveController* má na starost trénování kognitivních akcí, nastavení těchto akcí pro připojený headset a citlivost při rozpoznávání kognitivních akcí. Třída, která obsahuje funkce pro získání změny polohy headsetu a případně její resetování, se nazývá *GyroController*. Všechny tyto kontrolery jsou volány v implementaci třídy *MainController*, která obsahuje logiku celé aplikace. Kromě volání ostatních kontrolerů třída *MainController* má jako člena ukazatel na třídu *QTimer*. Třída *QTimer* je implementována v Qt framework a slouží jako časovač. Jakmile vymezený čas jedné iterace vyprší, je vyvolán signál, který je v konstruktoru třídy *MainController* spojený s funkcí *getNextEvent()*, definovanou jako slot. Ukázku kódu tohoto spojení můžeme vidět zde 2 a implementaci metody *getNextEvent()*, která je hlavní metodou aplikace, zde 4.

```

//MainController.h
public slots:
    void getNextEvent();
private:
    QTimer *m_mainTimer;
//MainController.cpp
m_mainTimer = new QTimer(this);
connect(m_mainTimer, SIGNAL(timeout()), this, SLOT(getNextEvent()));
  
```

Výpis 2: Deklarace slotu a jeho napojení na implementovaný signál ve třídě *QTimer*.

Třída `MainController` obsahuje ukazatel na člena třídy `Sender` a volá jeho funkce. Třída `Sender` patří do druhé skupiny, která už nepracuje s metodami hlavičkových souborů `Emotiv SDK`, a má vlastnost, která přijímá přeložené kognitivní akce, jejich přesnost a pohyb hlavy. Ty pak odesílá po síti jako broadcast ve formě UDP packetu, na portu 7777. Všechny takto odeslaná data jsou uložena v externím souboru "dataSent.log". Třetí skupina má jednu třídu `Emotiv2Robot`, která je potomkem třídy `QMainWindow`. Tím je umožněno vyvolat okno, do kterého je vykresleno GUI vytvořené v aplikaci Qt Creator. Další účel třídy `Emotiv2Robot` je propojit jednotlivé signály tlačítek a událostí z GUI k metodám kontrolerů a signály z kontrolerů s funkcemi, definovanými jako slot, s objekty v GUI pro jejich aktualizaci a zpětnou vazbu pro uživatele. Popsané vztahy mezi třídami s jejich vlastnostmi a členy můžeme vidět na diagramu tříd 16.

6.3.2 MindBot RobotPart

Aplikace `MindBot RobotPart` je napsaná v programovacím jazyce Java. Pro komunikaci s řídicí jednotkou EV3 a připojeným příslušenstvím bylo použito `LeJOS API`. Výchozí třída aplikace `RobotPart` je `RobotPart`. V této třídě probíhá inicializace tříd aplikace a jednotlivých motorů, což bylo ukázáno v kapitole 4.2.5. Ve třídě `RobotPart` je vytvořeno a spuštěno nové vlákno třídy `EmotivConnector`. Zde běží cyklus, který poslouchá na portu 7777, na broadcast datagramy. Přijaté datagramy přeloží do textového formátu, vykreslí na obrazovku robota a zavolá funkci instance třídy `Decoder`, kde předává přijatou zprávu v parametru. Ve třídě `Decoder` je zpráva přeložena na instrukce robotovi. Tyto instrukce jsou pak posílány v parametrech volaných funkcí třídy `Executor`, která implementuje rozhraní `Movements`, tedy je zde potřeba implementovat funkce pohybů robota. Implementace těchto funkcí obsahuje volání funkcí třídy `MotorCommand`. `MotorCommand` má jako členy jednotlivé motory a senzor. Je zde proto naprogramováno volání funkcí motorů, kterou jsou volány třídou `Executor`. Příklad příkazu zatočení můžeme vidět zde na výpisu kódu 3. Kód obsahuje implementaci funkce zatočení. Pilot je objekt třídy `MovePilot`, který je obsažen v `LeJOS API` a obsahuje diferenciální podvozek, který jsme nadefinovali v konstruktoru třídy `MotorCommand`. První se zkontroluje, jestli nějaký pohyb už neprobíhá, a poté se vypočítá poloměr a úhel. Tyto hodnoty jsou poté použity jako parametry funkce `arc`. Funkce `arc` vyvolá pohyb robota podle spočítaného kruhového oblouku. Ten se vypočítá podle vzorce $L = \frac{2\theta\pi r}{360^\circ}$. Vztahy mezi jednotlivými třídami, členy tříd a vlastnosti tříd můžeme vidět na diagramu tříd 17.

```
//MotorCommand.java
public void rotateDifferently (int angle, directions direction){
    if (!pilot.isMoving()){
        double r = 12.5 - ((angle / 360.0) * (12.5));
        double a = angle / 4;

        if (direction == directions.LEFT){
            if (r == 0){
                a = a * -1.0;
            } else{
                r = r * -1.0;
            }
        }
    }
}
```

```

    }
    this.pilot.arc(r, a, true);
  }
}

```

Výpis 3: Implementace funkce zatočení ve třídě Executor.

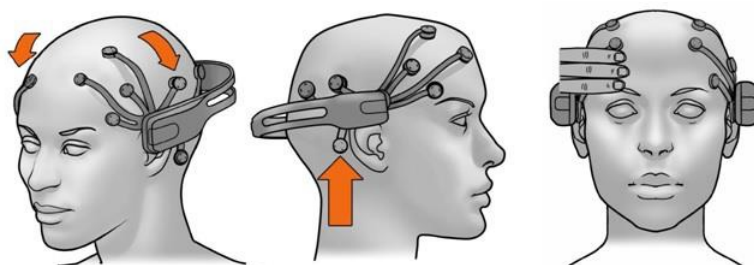
6.3.2.1 Command pattern Výše popsanou implementaci můžeme vyložit také z pohledu návrhového vzoru Command pattern, který byl použit. Jako *reciever* jsou použity třídy `EmotivConnector` a `Decoder`. Třída `Decoder` dekóduje zprávy ze sítě. Funkce třídy `MotorCommand`, která slouží jako *command*, jsou ihned po dekódování zprávy vyvolány třídou `Executor`, která slouží jako *invoker*. Proto není třeba explicitně vytvářet třídu pro funkci *client*, která v rámci návrhového vzoru Command pattern běžně slouží jako rozhodující článek, kdy se jaký příkaz vykoná. Dalo by se říct, že třída `Decoder` splňuje zároveň funkci *client*, protože rozhoduje, kdy se daný příkaz zavolá. To je v případě aplikace `MindBot RobotPart` ihned po dekódování zprávy.

6.4 Postup používání

6.4.1 Požadavky a předpoklady

Níže v tabulce 2 je uvedený stručný seznam zařízení, softwaru a dovedností, kterých bylo potřeba k vypracování. Všechna uvedená zařízení byly vypůjčeny od Vysoké školy báňské, díky které bylo tento projekt možné uskutečnit.

Bylo třeba naučit se pracovat s klíčovými přístroji EEG headsetem EPOC a řídicí jednotkou `Lego Mindstorms EV3`, vytvořit model robota, který byl postaven ze sady stavebnice `Lego Mindstorms` kompatibilní s `EV3`. Robot musí být schopný pohybu dopředu, dozadu, doleva a doprava, což je jeden ze základních předpokladů této práce. Dále nedílnou součástí je udržení myšlenek pro 4 natrénované kognitivní akce snímané EEG headsetem. Bylo důležité také porozumět EEG signálům a kognitivním funkcím, které mají svá centra uložena v různých částech mozku.



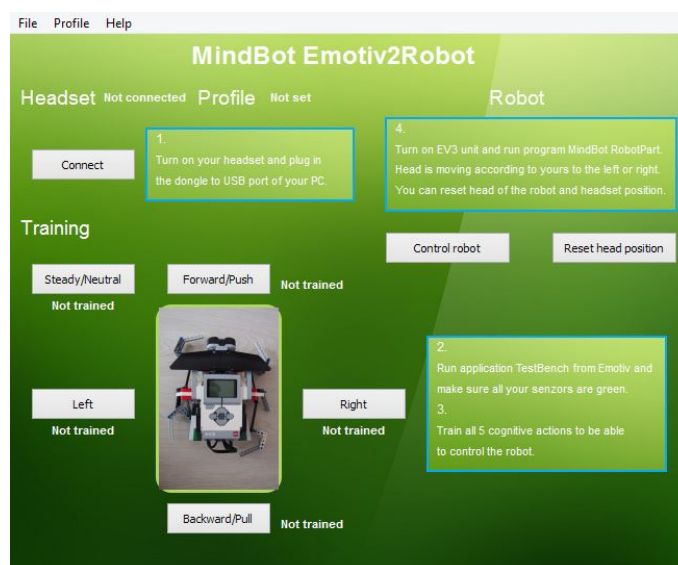
Obrázek 8: Správné nasazení headsetu [7].

Headset EPOC	Umět kontrolovat 4 kognitivní akce
PC	2.4 GHz Intel Pentium 4 processor Windows 7, Windows 8 nebo Windows 10 2GB RAM 200MB místa na disku 1x USB 2.0 port Emotiv SDK Microsoft Visual Studio, Qt framework, C++ Eclipse IDE, LeJOS Program Emotiv TestBench
Sada Lego Mindstorms EV3	Přepsat firmware řídící jednotky EV3 na LeJOS firmware Sestrojený model robota Edimax N150 Wi-Fi Nano USB Adaptér
Ostatní znalosti	Porozumět EEG a částí mozku

Tabulka 2: Tabulka zobrazující přehled nezbytného hardwaru, softwaru a znalostí

6.4.2 Příprava

Připojte svůj PC a řídící jednotku EV3 do stejné WiFi sítě. Pokud nemáte žádnou dostupnou WiFi síť, vytvořte si přístupový bod na svém PC. Dále je třeba navlhčit jednotlivé elektrody solným roztokem. Nejlépe uděláte, když při navlhčování necháte elektrody v pouzdru. Jakmile jsou elektrody navlhčené, připojte je k headsetu. Nasad'te si headset podle obrázku 8 a připojte USB přijímač od Emotiv do USB portu vašeho počítače. Spusťte si program Emotiv TestBench, na kterém zkontrolujete kontakt jednotlivých diod. Je zapotřebí, aby všechny diody signalizovaly zelenou barvu, případně žlutou. Příklad použití programu TestBench můžeme vidět na obrázku 18. Po úspěšném nasazení a zkontrolování kontaktu elektrod headsetu spusťte program MindBot RobotPart pomocí menu na řídící jednotce robota. Pokud řídící jednotka neobsahuje program RobotPart, nahrajte jej na microSD kartu, která je zasunuta v řídící jednotce EV3, a microSD kartu vložte zpět. Po spuštění programu RobotPart řídící jednotka začne oranžově blikat a napíše na obrázku, že čeká na příkazy.



Obrázek 9: Úvodní obrazovka aplikace MindBot Emotiv2Robot.

6.4.3 Ovládání robota

Po úspěšné přípravě headsetu a robota spusťte aplikaci MindBot Emotiv2Robot na svém počítači a postupujte podle pokynů zobrazených v aplikaci. Grafické prostředí aplikace Emotiv2Robot je na obrázku 9. Připojte headset pomocí tlačítka *Connect* a vyberte nebo vytvořte si svůj profil. Po zvolení profilu je třeba natrénovat 4 kognitivních akce a neutrální stav kliknutím na tlačítko dané akce. Trénování trvá přibližně 8 sekund a aplikace Emotiv2Robot vám oznámí začátek a konec tréninku, případně pokud došlo k nějaké chybě, a je tedy nutné provést trénink znovu a zkontrolovat spojení elektrod v aplikaci TestBench. Jakmile máte všechny kognitivní akce natrénované, můžete začít ovládat robota pomocí tlačítka *Control robot* a robot by měl přestat oranžově blikat a začít svítit zeleně. Kromě pohybu dopředu, dozadu, doleva a doprava je možné pohybovat hlavou robota, a to tak, že otočíte vaši hlavu doleva či doprava. V případě, že vaše hlava a hlava robota bude v jiné pozici, použijte tlačítko *Reset head position*, které vyresetuje pozici hlavy robota na střed a vynuluje pozici headsetu. Při ukončení aplikace se váš natrénovaný profil automaticky uloží do souboru *profile.bin*, z kterého se bude načítat natrénovaný model v případě opakovaného použití. Změnit profil můžete za běhu aplikace, a to z vrchního menu pod možností *Profile->Setup profile*.

7 Experimenty

Pro provedení experimentů bylo třeba zvolit co nejefektivnější sadu myšlenek či úkonů pro natrénování kognitivních akcí. Na základě studií uvedených v kapitolách 7.1 a 7.2, bylo zvoleno trénování kognitivních akcí pomocí vnitřního monologu v jiném než rodném jazyce. Jednalo se o jedno slovo pro jeden druh pohybu, které souviselo s tím daným směrem pohybu. Zapojena byla také vtělená kognice pohybu ruky daným směrem, kterým se měl robot pohybovat, pro jednodušší vyvolání a přesnější rozpoznání kognitivních akcí.

7.1 Různé jazyky

Jazyk je jedním z nejdůležitějších prostředků při využívání mozku, a přesto velmi málo je známo o tom, jak je interpretován. Víme, že pro psaný jazyk má pravá strana mozku tendenci interpretovat většinu tohoto úkonu. Ale pro mluvené slovo se zdá být interpretace pro mozek mnohem složitější, protože musí zároveň odfiltrovat náhodné zvuky a případně některým z nich dát smysl. Pro vnitřní monolog máme ještě méně dostupných informací než u mluveného slova [17].

V jednom z uvedených projektů 2.3 bylo zjištěno, že tato metoda je jedna z nejpresnějších. Vybavování si slova v hlavě z jiného než rodného jazyka se aktivovala asociální somatosenzitivní oblast, která je vzdálená natolik od ostatních snímaných broadbandových okrsků headsetem EPOC, že headset dokáže signál lépe rozpoznat mezi ostatními a tím přesněji ovládat model robota.

7.2 Vtělená kognice

Vtělená kognice představuje v kognitivní vědě výzkum, který klade důraz na důležitost formativní role prostředí při vývoji kognitivních procesů. Obecná teorie tvrdí, že kognitivní procesy se vyvíjejí, když se z interakcí, které jsou orientované na cíl a probíhají v reálném čase mezi organismy a jejich prostředím, vynoří propojené systémy. Povaha těchto interakcí ovlivňuje tvorbu a blíže specifikuje charakter vývoje kognitivních kapacit [23].

Nejčastějším příkladem je pokus „tužka v zubech“. Účastníci tohoto experimentu drží v zubech tužku, čímž zaměstnávají svaly, které používáme při úsměvu. Tito lidé rozpoznávají příjemné věty mnohem rychleji než ty nepříjemné. Naopak držení tužky mezi nosem a ústy zaměstnává svaly, které používáme při mračení – efekt na rozpoznávání vět je opačný. Vědci věří, že toto reprezentuje vtělenou kognici. Kritici však tvrdí, že výsledný efekt tohoto experimentu byl znepřesněn užitím určitých (usnadňujících) obličejových svalů [24].

7.3 Odezva mezi aplikacemi

Byl proveden test odezvy pohybu robota na programem vyvolanou kognitivní akci. Byla proto použita aplikace EmoComposer, která byla vyvinuta společností Emotiv. EmoComposer dokáže vyslat signál, který vypadá jako definovaná kognitivní akce se zvolenou si-

lou kognitivní akce. Výsledek tohoto testu bude použit pro upřesnění metodiky použité při následujícím testu, u kterého budou kognitivní akce vyvolané lidmi. Měl by ukázat relevantnost zpoždění vytvořené aplikacemi s dobou pro vyvolání kognitivní akce uživatelem.

Byl stopován čas od kliknutí na vyslání signálu z EmoComposer a čas, kdy se robot rozpohyboval. Interval volání funkce pro zjištění nového stavu z EmoEngine byl v aplikaci Emotiv2Robot nastaven na 100 ms. Zpočátku bylo měření času provedeno pomocí aplikace Stopwatch na iOS zařízení, ale protože odezva byla téměř nulová, bylo potřeba provést přesnější měření, a to pomocí zaznamenání akce s časem do souboru dataSent.log. Z 1000 získaných záznamů byl medián odezvy 126 ms. Což je pouze o 26 ms více než interval pro zjištění změny stavu získaného z EmoEngine. Můžeme tedy tvrdit, že odezva mezi aplikacemi nemá vliv na odezvu při pokusu rozpohybovat robota pomocí headsetu EPOC. Kdybychom si vzali extrémní případ jako řízení formule 1. Na základě studie [29], je reakční doba řidiče formule 1 mezi 0,25 až 0,3 sekundy.

7.4 Experiment přesnosti a odezvy při ovládání robota headsetem

Tento experiment se zaměřuje na přesnost ovládání a prodlevu mezi vyvoláním zadané kognitivní akce pro pohyb robota. Sada kognitivních akcí pro natrénování byla vybrána na základě poznatků zmíněných na začátku kapitoly 7 a společného cizího jazyku pro všechny účastníky, kterým byla angličtina. Zadáním bylo, udržet a vyvolat co nejvíce krát jeden druh kognitivní akce po dobu 1 minuty a nesplést se. Postupně byly testovány všechny 4 akce u všech 5 účastníků v tomto daném pořadí: dopředu, dozadu, doleva a doprava. Podařilo se získat účastníky s rozdílnými rodnými jazyky, tím se zajistilo, aby výchozí rodný jazyk neměl vliv na celkový výsledek a v případě velké odchylky, zjištění, že výchozí rodný jazyk má vliv na tuto sadu kognitivních akcí. Účastníci experimentu byli 4 muži a 1 žena ve věkovém rozmezí mezi 26 a 44 lety. Tito účastníci neměli předchozí zkušenost s používáním headsetu nebo robotikou a mechanismy v robotice.

Experiment probíhal následovně. Testovaný subjekt se usadil a bylo mu popsáno zadání experimentu. Testovaný si sám mohl vybrat 4 slova v angličtině, která se mu nejvíce spojují se 4 možnými pohyby robota. Je vyžadováno, že při trénování jednotlivých akcí by měl zkoušet pohnout pravou rukou ve směru pohybu robota. Po představení zadání byl nasazen subjektu headset a pomocí aplikace Emotiv TestBench bylo zkontrolováno navázání spojení mezi elektrody a hlavou účastníka experimentu. Následoval trénink těchto předem připravených akcí. Bylo třeba natrénovat všech 5 akcí, kdy při trénování neutrální akce se účastník experimentu snažil sedět v klidu, bez mluvení a žádných dalších úkonů. Potom bylo otestováno v průběhu 5 minut, že účastník dokáže vyvolat každou natrénovanou kognitivní akci. Potom postupně v tomto pořadí dopředu, dozadu, doleva a doprava bylo testováno vyvolání těchto kognitivních akcí, po dobu 1 minuty. V aplikaci Emotiv2Robot byla nastavena nejvyšší citlivost rozpoznávání kognitivních akcí, což přineslo více chyb, ale za to zvýšilo celkový počet vyslaných příkazů.

Na obrázku 19 je tabulka, zobrazující výsledky experimentu. Po sečtení hodnot získáváme celkovou úspěšnost, která je 67,14% mezi všemi účastníky a průměrnou dobou prodlevy pro zopakování vyvolání akce bylo 17,225 sekund. Při provádění expe-

rimentu bylo zaznamenáno, že někteří účastníci se pokoušeli zavřít oči pro zvýšení koncentrace a přijímání méně rušících okolních vjemů. Účastník Pavol tuto metodu zkoušel a dokázalo se, že zavírat oči spíše uškodí. Nezpůsobilo to zmenšení chybovosti, ani nezlepšilo prodlevu mezi vyvoláním stejné akce znovu. Vypozorovalo se, že jakmile účastník experimentu zaznamená pohyb robota ve správném směru, automaticky se snaží vyvolat stejnou myšlenku, při které tento úkon byl vyvolaný. Dalo by se říct, že člověk automaticky ve svém podvědomí aktivuje učící algoritmus tím, že se snaží vyvolat předchozí kognitivní myšlenku, která vyvolala zadaný pohyb robota. Tento učící mechanismus přinášel, také částečně negativní dopad na výsledek testu. Docházelo k odklonění od zadaného slova ze sady kognitivních akcí, ale jakmile účastník experimentu zjistil, že nedokáže vyvolat danou myšlenku pomocí zopakování, vrátil se k použití slova, které úspěšně vyvolávalo pohyb robota.

8 Závěr

Propojení robota s headsetem EPOC a jeho řízení bylo úspěšně implementováno. V rámci aplikace Emotiv2Robot, která slouží jako grafické prostředí, umožňující vytvoření a uložení profilu, ke kterému jsou trénovány kognitivní akce a je vytvořena komunikace s robotem. Aplikace odesílá informace, které umožňují pohyb robota do 4 stran, který se může lišit na základě síly kognitivní akce. Informace obsahují také pozici headsetu, která je odeslána aplikaci běžící na robotovi, a ten díky toho dokáže otáčet hlavou doleva a doprava, podle toho, jak uživatel pohybuje svou hlavou. Zvolené 4 kognitivní akce možná nebyli správným řešením, protože přesnost ovládání robota byla 67,14%, což nemůžeme považovat jako dostatečné procento, například pro řízení auta. Na druhou stranu, tato data byla získána v rámci experimentu, v kterém se podíleli uživatelé, kteří pracovali s headsetem dohromady půl hodiny. Opakovaným zkoušením se přesnost ovládání zvyšovala, ale bohužel při 4 kognitivních akcích ani zkušený uživatel nedosáhl přesnosti nad 90%. Headset EPOC je schopný přijímat surová data a provádět nad nimi optimalizaci. Optimalizace a práce se surovými daty je součástí rozšířeného SDK vyvíjeného na VŠB. Tento způsob nebyl zvolen a mohl by být použit jako rozšířené řešení této práce, pro srovnání efektivnosti rozpoznání natrénovaných akcí pomocí Emotiv SDK. Aplikace Emotiv2Robot se dá použít s jakýmkoli zařízením, které by zpracovávalo zprávy, které tato aplikace posílá. Například vozíčkové křeslo, které bude mít ARM procesor a bude na něm běžet Linux distribuce s Java Virtual Machine. V takovém případě by se mohl použít dokonce stejný programovací jazyk a firmware, jako byl použit v této práci. Zpracováním tohoto projektu jsem získal znalosti o EEG, seznámil se s headsetem EPOC a jeho SDK, naučil se pracovat s novým firmware LeJOS při programování řídicí jednotky Lego Mindstorms EV3 a naučil se používat rozšířením pro C++, Qt.

Jan Beseda

9 Reference

- [1] Wired Magazine. Pollack, M. *Mind control: How a £200 headset is redefining brain-computing interaction* [online]. Vystaveno roku 2010 [cit. 2015-07-15]. Dostupné na: <<http://www.wired.co.uk/magazine/archive/2010/12/features/mind-control>>.
- [2] Emotiv. Insight, *Emotiv Insight - a sleek, 5-channel, wireless headset* [online]. Vystaveno roku 2014 [cit. 15. 07. 2015]. Dostupné na: <<https://emotiv.com/insight.php>>.
- [3] Katedra zoologie a ornitologie, PřF UP Olomouc. RNDr. Vlasta Lungová Ph.D. *Stavba a funkce lidského mozku* [online]. Vystaveno roku 2012 [cit. 06. 04. 2016]. Dostupné na: <<http://pfyziolfup.upol.cz/castwiki/?p=3265>>.
- [4] Computational and Mathematical Methods in Medicine. PubMed, *Hypovigilance Detection for UCAV Operators Based on a Hidden Markov Model* [online]. Vystaveno roku 2014 [cit. 06. 04. 2016]. Dostupné na: <https://www.researchgate.net/publication/263394838_Hypovigilance_Detection_for_UCAV_Operators_Based_on_a_Hidden_Markov_Model>.
- [5] Emotiv, *Emotiv Software Development Kit User Manual [manuál k přístroji]* [online]. Vystaveno roku 2009 [cit. 03. 07. 2015]. Dostupné na: <emotiv.com/developer/SDK/UserManual.pdf>.
- [6] Emotiv. EPOC, *A revolutionary Brain Computer Interface and Scientific Contextual EEG offering high resolution, 14 EEG channels + 2 references* [online]. Vystaveno roku 2011 [cit. 15. 07. 2015]. Dostupné na: <<https://emotiv.com/epoc.php>>.
- [7] Emotiv. EPOC průvodce, *Průvodce k použití* [online]. Vystaveno roku 2011 [cit. 15. 07. 2015]. Dostupné na: <<https://emotiv.com/quickstart-guides/QuickStartGuide2014.pdf>>.
- [8] Wiki Emotiv. SDK, *Používání API ke komunikaci s EmoEngine* [online]. Vystaveno roku 2011 [cit. 15. 07. 2015]. Dostupné na: <<http://wiki.emotiv.com/tiki-index.php?page=API+Example>>.
- [9] VSB. Emotiv SDK, *Signal Data Processing* [online]. Vystaveno roku 2013 [cit. 20. 07. 2015]. Dostupné na: <<http://www.cs.vsb.cz/gajdos/projects.php>>.
- [10] Botbench. Xander, *Comparing the NXT and EV3 bricks* [online]. Vystaveno roku 2013 [cit. 20. 07. 2015]. Dostupné na: <<http://botbench.com/blog/2013/01/08/comparing-the-nxt-and-ev3-bricks/>>.
- [11] Wikipedia, *leJOS firmware* [online]. Vystaveno roku 2015 [cit. 20. 07. 2015]. Dostupné na: <<https://en.wikipedia.org/wiki/LeJOS>>.
- [12] Wikipedia, *Lego Mindstorms series of kits contain software and hardware to create customizable, programmable robots* [online]. Vystaveno roku 2015 [cit. 20. 07. 2015]. Dostupné na: <https://en.wikipedia.org/wiki/Lego_Mindstorms>.

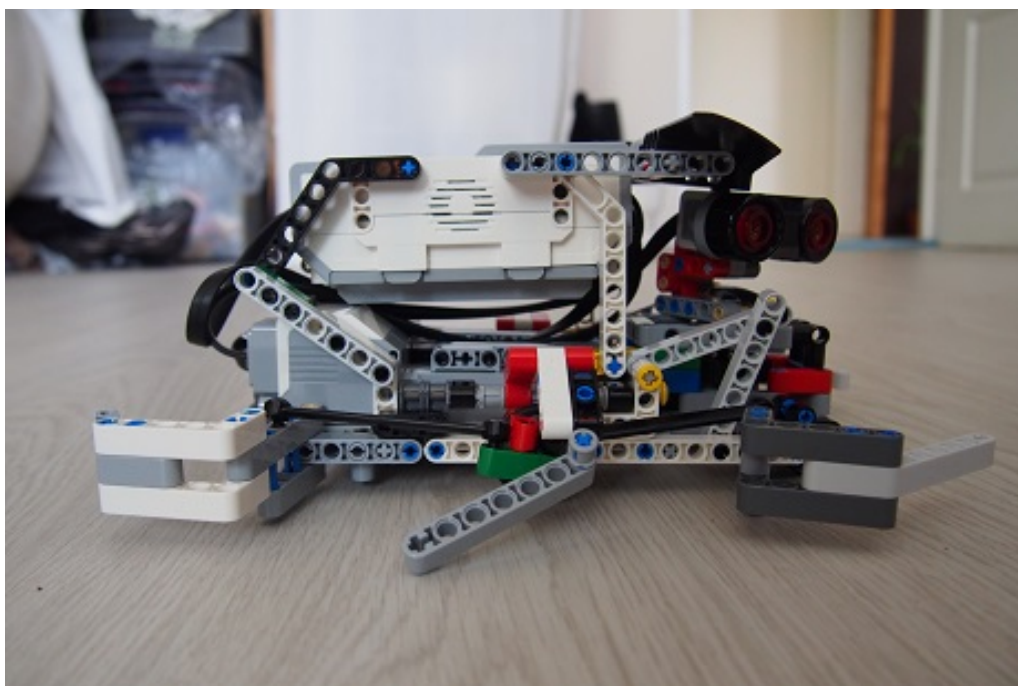
-
- [13] Carl Gustav Calwer; Gustav Jäger *Naturgeschichte der Käfer Europas*. Stuttgart: Julius Hoffmann, (1876) Käferbuch. [online] Dostupné na: <www.biolib.de>
 - [14] Lego. Mindstorms, *SPIK3R six-legged creature*. [online]. Vystaveno roku 2013 [cit. 20. 07. 2015]. Dostupné na: <<http://www.lego.com/en-us/mindstorms/build-a-robot/spik3r>>.
 - [15] Lego. Shop, *Lego e-shop* [online]. Vystaveno roku 2013 [cit. 20. 07. 2015]. Dostupné na: <<http://shop.lego.com/>>.
 - [16] A. Vourvopoulos a Fotis Liarokapis, *Robot Navigation using Brain-Computer Interfaces*. iWARG. Coventry University, UK. 2012
 - [17] EPOC-alyipse. Christopher Perez; Lee Sully; Kathryn Morales; Michael Strobridge *Mind Controlled RC Car* [online]. Vystaveno roku 2013 [cit. 03. 04. 2016]. Dostupné na: <<http://www.eecs.ucf.edu/seniordesign/fa2012sp2013/g21/FinalPaperII.pdf>>.
 - [18] WikiSkripta. Mefanet, *Elektroencefalografie* [online]. Vystaveno roku 2016 [cit. 03. 04. 2016]. Dostupné na: <<http://www.wikiskripta.eu/index.php/Elektroencefalografie>>.
 - [19] WikiSkripta. CabiCz, *Tabulka aktivít EEG* [online]. Vystaveno roku 2011 [cit. 06. 04. 2016]. Dostupné na: <http://www.wikiskripta.eu/index.php/Soubor:Tabulka_otazka_243.JPG>.
 - [20] Ústav Normální Anatomie. Lékařská fakulta Univerzity Palackého v Olomouci, *NEUROANATOMIE I. - Struktury CNS, Telencephalon; Cerebrum* [online]. Vystaveno roku 2008 [cit. 06. 04. 2016]. Dostupné na: <<http://www.nan.upol.cz/neuro/cd699.html>>.
 - [21] Univesity of Michagan, *Broadmann areas* [online]. Vystaveno roku 2008 [cit. 07. 04. 2016]. Dostupné na: <<http://umich.edu/cogneuro/jpg/Brodmann.html>>.
 - [22] Mozková kůra, David Kachlík *TELENCEPHALON, CNS, Mozková kůra* [online]. Vystaveno roku 2015 [cit. 07. 04. 2016]. Dostupné na: <http://anatomie.lf3.cuni.cz/centralni_prezentace/Regulace1_mozkova_kura.pdf>.
 - [23] Embodied Cognition, Monica Cowart *Merrimack College* [online]. [cit. 08. 04. 2016]. Dostupné na: <<http://www.iep.utm.edu/embodcog/>>.
 - [24] Embodied cognition, Adams, F. *Phenomena of Cognitive Science* 9 (4), doi:10.1007/s11097-010-9175-x.. Vystaveno roku 2010 [cit. 08. 04. 2016].
 - [25] Department of Signal Theory, Communications and Telematics Engineering, University of Valladolid, Jaime Gomez-Gil, Israel San-Jose-Gonzalez, Luis Fernando Nicolas-Alonso, a Sergio Alonso-Garcia., *Steering a Tractor by Means of an EMG-Based Human-Machine Interface* [online]. Vystaveno roku 2011 [cit. 12. 04. 2016]. Dostupné na: <<http://www.mdpi.com/1424-8220/11/7/7110>>.

-
- [26] College of Engineering, Wichita State University, Eric Berman, Kalyn Kovac a Bruno Umeadi. *Mind-controlled R/C Car* [online]. Vystaveno roku 2012 [cit. 12. 04. 2016]. Dostupné na: <<http://cratel.wichita.edu/blogs/eecsfinalreportspr2012mindcontrolledrccar/>>.
- [27] Interactive Worlds Applied Research Group, Coventry University, Athanasios Vourvopoulos a Fotis Liarokapis. *Brain-controlled NXT Robot: Tele-operating a robot through brain electrical activity* [online]. Vystaveno roku 2011 [cit. 12. 04. 2016]. Dostupné na: <<http://www.fi.muni.cz/liarokap/publications/VSGAMES2011a.pdf>>.
- [28] Qt Documentation, Qt. *Signals and Slots* [online]. Vystaveno roku 2016 [cit. 16. 04. 2016]. Dostupné na: <<http://doc.qt.io/qt-4.8/signalsandslots.html>>.
- [29] Autolexicon, Ing. Jan Sajdl, Ph.D. *Reakční doba řidiče* [online]. Vystaveno roku 2011 [cit. 16. 04. 2016]. Dostupné na: <<http://www.autolexicon.net/cs/articles/reakni-doba-ridice/>>.

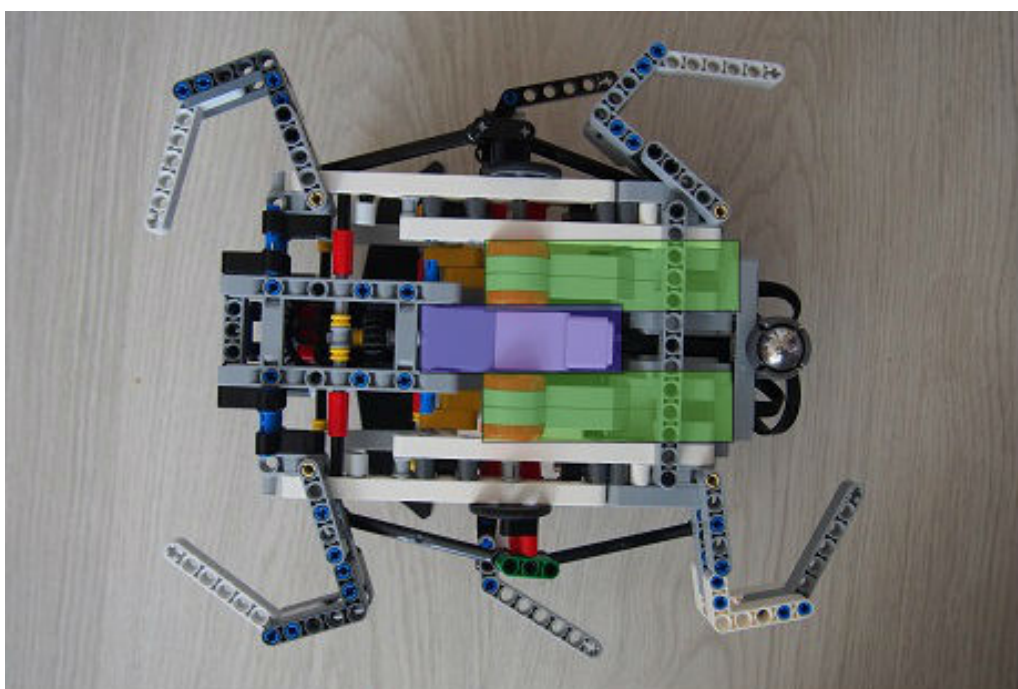
A Obrazy



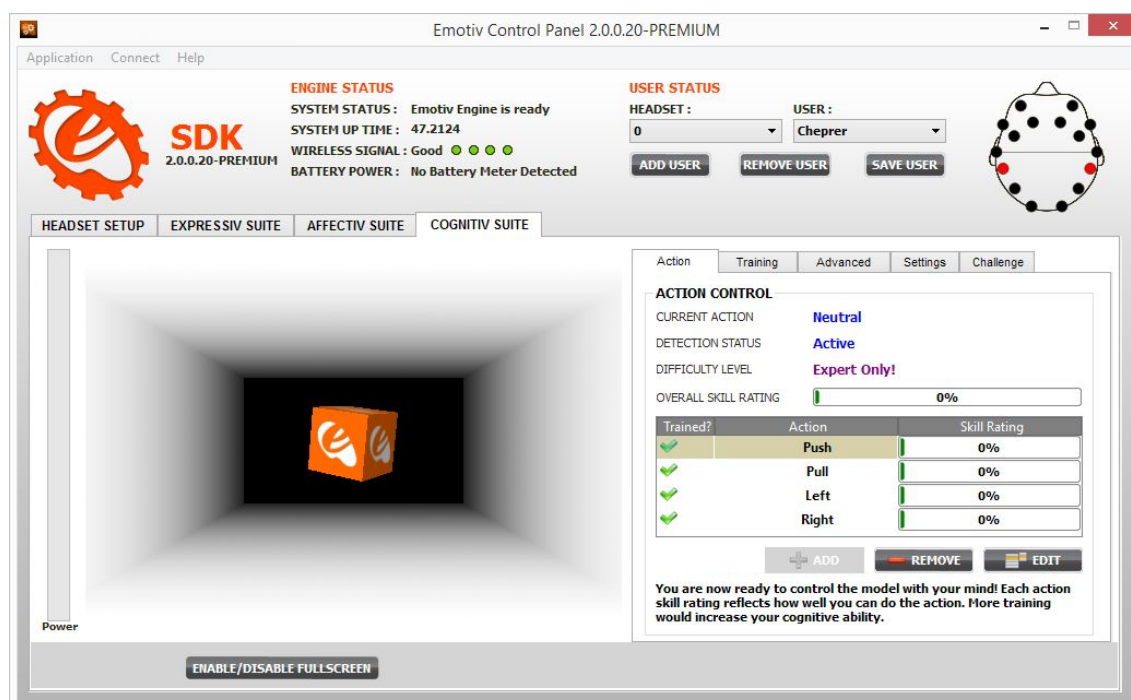
Obrázek 10: Cheprer, model robota, z pohledu ze shora.



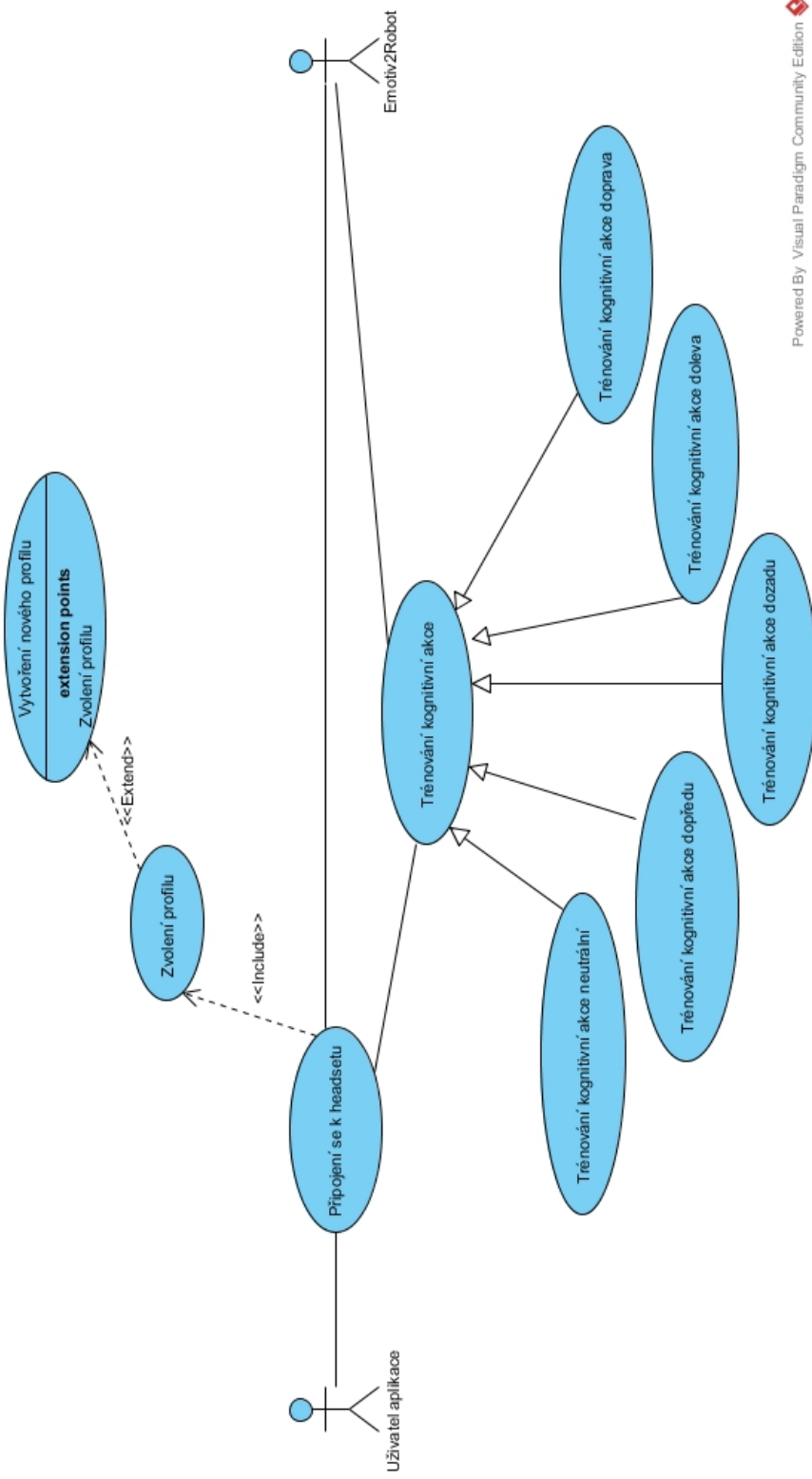
Obrázek 11: Cheprer, model robota, z pohledu z pravého boku



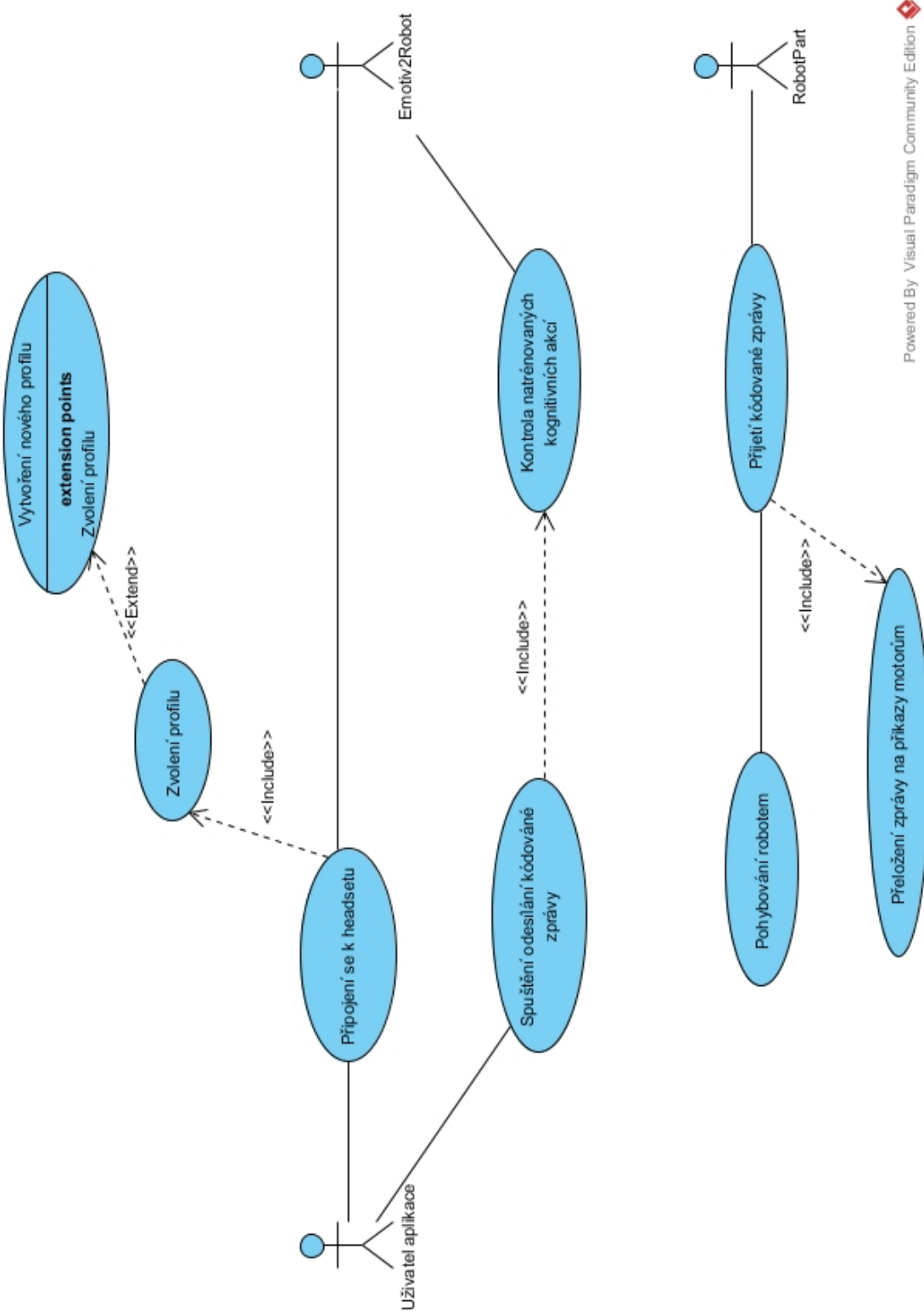
Obrázek 12: Cheprer, model robota, z pohledu zespodu. Na konstrukci jsou zeleně vyznačeny EV3 Large Servo Motor a modře EV3 Medium Servo Motor.



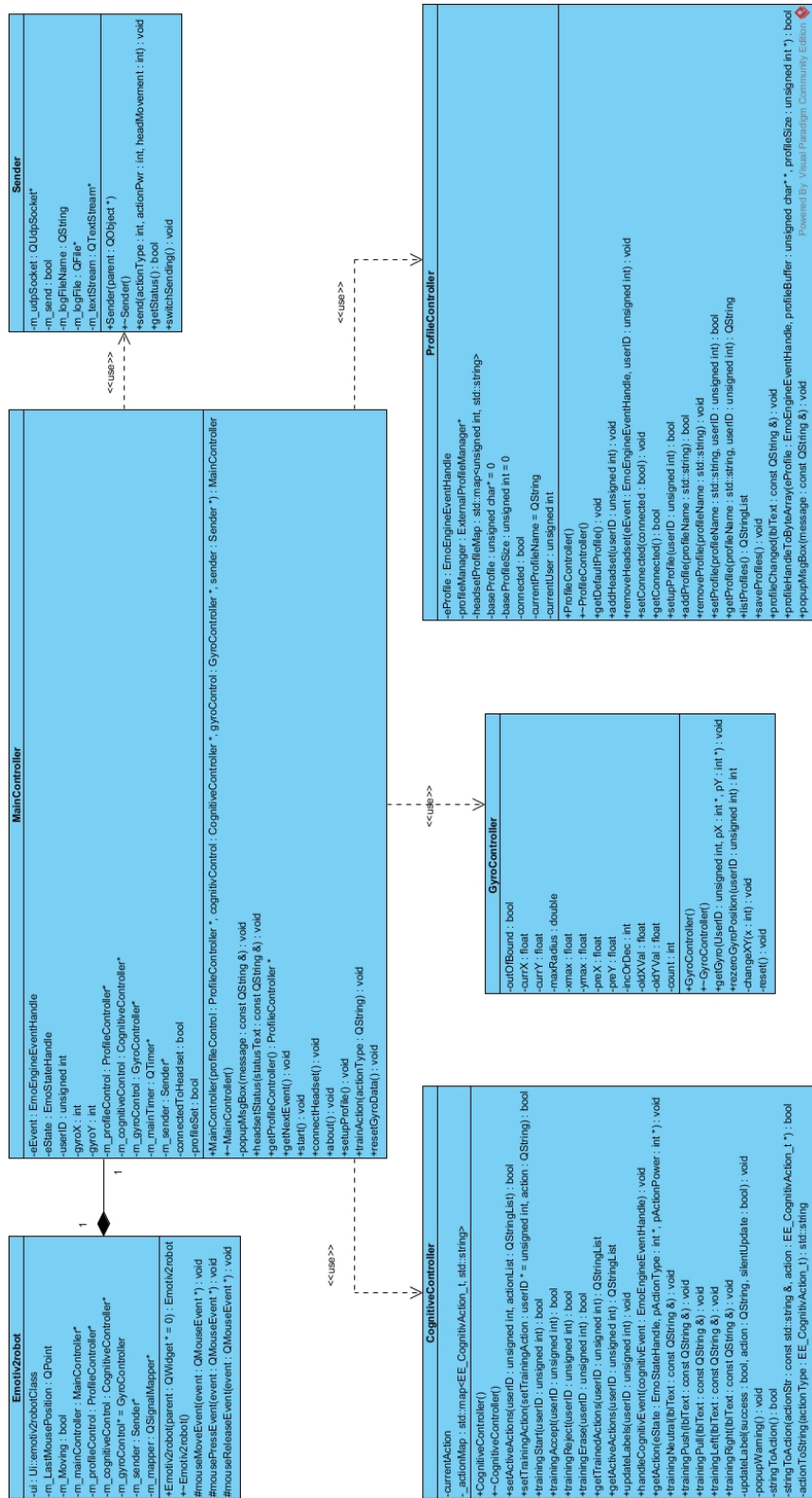
Obrázek 13: Aplikace Emotiv Control Panel - Cognitiv Suite s připojeným headsetem a bez nasazení.



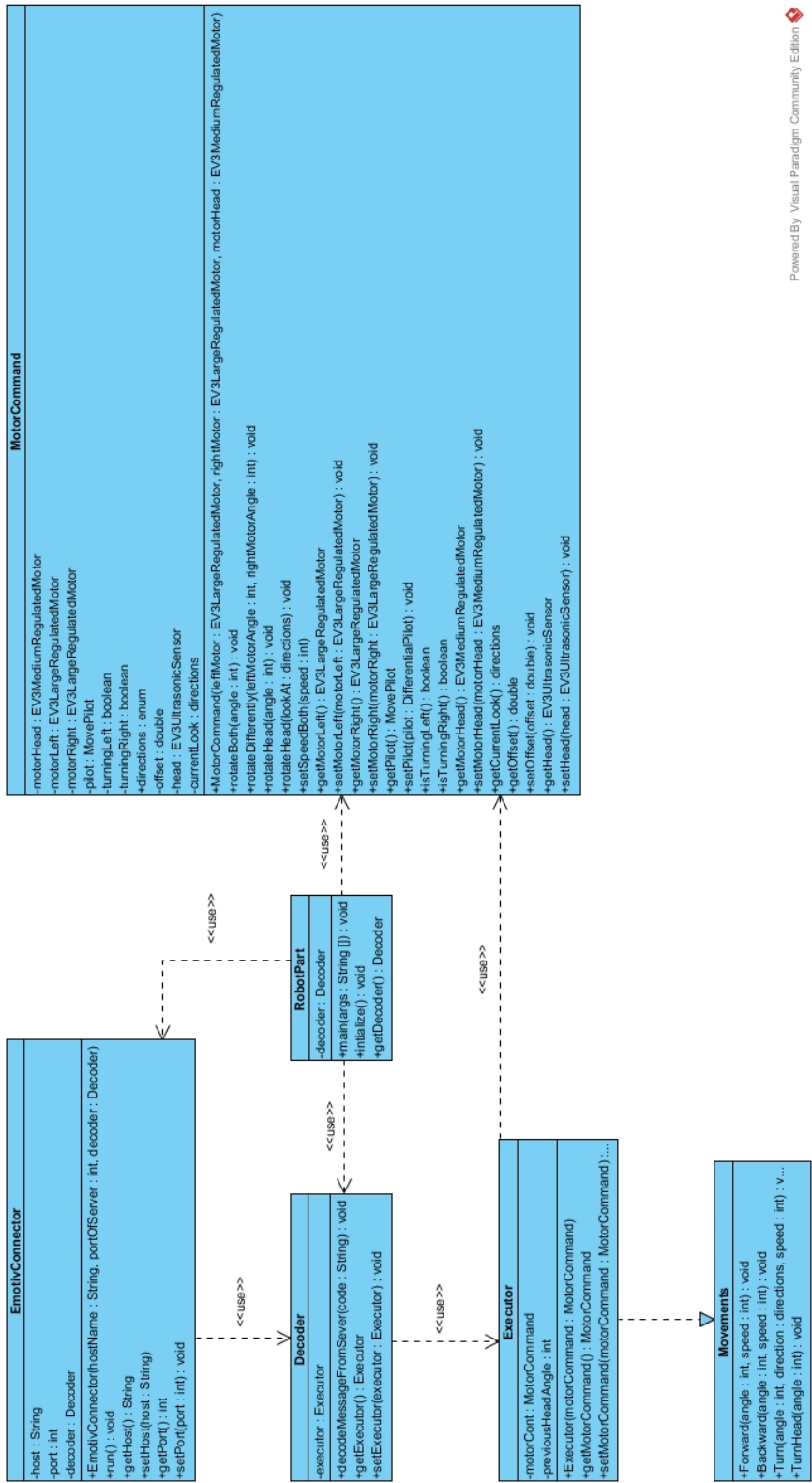
Obrázek 14: Diagram případu užití zobrazuje případ, kdy aktér, uživatel headsetu, trénuje jednotlivé kognitivní akce.



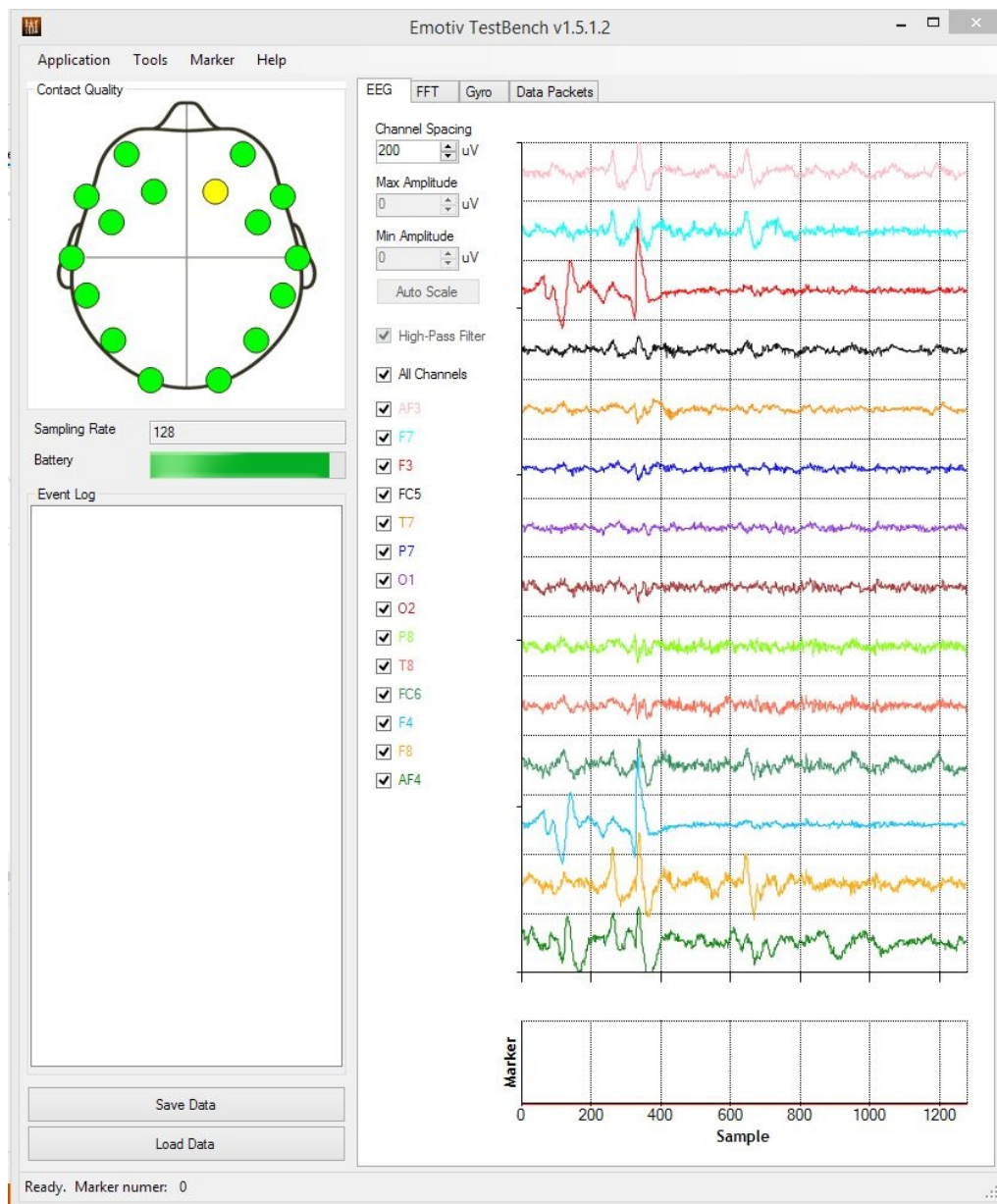
Obrázek 15: Digram případu užití ukazuje aktéra, uživatele headsetu, jak ovládá robota za pomoci aplikace Emotiv2Robot a aplikace RobotPart.



Obrázek 16: Třídní diagram aplikace MindBot Emotiv2Robot.



Obrázek 17: Třídní diagram aplikace MindBot RobotPart.



Obrázek 18: Aplikace Emotiv TestBench zobrazuje správný či špatný kontakt elektrod s hlavou uživatele a EEG aktivitu [7].

B Tabulky

Doleva		Doprava			
Počet vyvolání	Průměr prodlevy pro vyvolání(s)	Počet chybo	Počet vyvolání	Průměr prodlevy pro vyvolání(s)	Počet chybo
6	9,3	6	4	15	4
2	19	5	5	9,2	2
2	32	10	2	30	9
9	6,1	9	4	17,25	8
4	14,5	9	7	8	8
4,6	16,18	7,8	4,4	15,89	6,2

Obrázek 19: Tabulka zobrazující výsledky experimentu.

	EEG HEADSET
Počet kanálů	14 (plus CMS/DRL reference, P3/P4 lokace)
Názvy kanálů (tzv. 10-20 ale pouze 14)	AF3, F7, F3, FC5, T7, P7, O1, O2, P8, T8, FC6, F4, F8, AF4
Vzorkovací metoda	Sekveční vzorkování, jednoduché ADC
Vzorkovací frekvence	128 SPS (2048 Hz vnitřní)
Rozlišení	14 bitů 1 LSB = 0.51uV
Šířka pásma	0,2 - 45 Hz , digitální filtry vrubu 50Hz a 60Hz
Filtrování	Zabudovaný digitální Sinc filtr 5. pořadí
Dynamický rozsah	8400uV (pp)
Spojovací mód	AC spojený
Konektivita	Proprietární bezdrátové, 2.4GHz
Zdroj	LiPoly
Kapacita baterie	12 hodin
Měření impedance	Real-time kontaktní kvalita

Tabulka 3: Tabulka specifikací headsetu. [5]

C Výpisy kódu

```

// MainController.cpp
void MainController::getNextEvent(){
    int state = EE_EngineGetNextEvent(eEvent);

    if (state == EDK_OK) {

        EE_Event_t eventType = EE_EmoEngineEventGetType(eEvent);
        EE_EmoEngineEventGetUserId(eEvent, &userID);

        QStringList actionsToActivate;
        actionsToActivate << "push" << "pull" << "left" << "right";
        switch (eventType) {

            // Headset connected
            case EE_UserAdded:
            {
                m_profileControl->addHeadset(userID);
                emit headsetStatus("Connected");
                connectedToHeadset = true;
                m_profileControl->setConnected(m_profileControl->setupProfile(userID));
                if (m_profileControl->getConnected()){
                    m_cognitiveControl->setActiveActions(userID, actionsToActivate);
                    m_cognitiveControl->updateLabels(userID);
                }
                break;
            }

            // Headset disconnected
            case EE_UserRemoved:
            {
                m_profileControl->removeHeadset(eEvent, userID);
                emit headsetStatus("Not_connected");
                break;
            }

            case EE_EmoStateUpdated:
            {
                EE_EmoEngineEventGetEmoState(eEvent, eState);
                int actionType = 0, actionPwr = 0;

                m_cognitiveControl->getAction(eState, &actionType, &actionPwr);

                m_gyroControl->getGyro(userID, &gyroX, &gyroY);

                if (m_sender->getStatus()){
                    m_sender->send(actionType, actionPwr, gyroX);
                }
                break;
            }

            case EE_CognitivEvent:

```

```
    {
        m_cognitiveControl->handleCognitivEvent(eEvent);
        break;
    }

    default:
        this->popupMsgBox("Unhandled_operation_" + eventType);
        break;
    }
}
else if (state != EDK_NO_EVENT) {
    this->popupMsgBox("Internal_error_in_Emotiv_Engine!");
    this->m_mainTimer->stop();
}
}
```

Výpis 4: Implementace hlavní metody třídy MainController. Je volána každých 100ms.